RFTerm® Reference Guide

Notices

This software is installed in LXE computer equipment. Any reference, whether direct or implied, to any LXE equipment requires the reader to refer to the specific equipment User Guide for cautions, warnings and federal notices (e.g. FCC, EMC, UL, CE, etc.).

Revision Notice

Upgrade from Revision N to Revision P

Section	Explanation
Chapter 1 – Introduction	Added new section: "About RFTerm"
Chapter 2 – Quick Start	Added MX8 to "RFTerm Screens on Different Sized Platforms".
Chapter 4 – Session Configuration	Added new section: "HX2 Function Key Enable".

Appendix A – Key Maps	Added the following key maps:
	    Virtual key maps
	    HX2 key maps for new keypad styles
	    MX8 keymaps.
	Revised MX7 keymaps for Scroll Left One Row and Scroll Right One Row.

Table of Contents

Illustrations

Chapter 1 Introduction

Overview

RFTerm® allows users to connect to applications running on IBM® 3270, AS/400 and VT hosts from a Microsoft® Windows® based LXE® mobile computer over a wireless TCP/IP data network.

RFTerm can be used to directly communicate with host applications. Users can connect and log on to the host applications without the need for an intermediary device.

Assumptions

This reference guide assumes you have working knowledge of:

- Microsoft Windows user interface metaphor and terminology.

- Stylus based touch screen navigation terminology.

- Basic knowledge of concepts and terms for wireless networking.

- Basic operations and requirements of the host applications you want to access with LXE terminal emulation clients.

Conventions Used in this Guide

This guide uses the following typographical conventions:

- All user actions and interactions with the application are in bold, as in **Session | Configure**

- Any precautionary notes or tips are presented as in

    Tip: Text associated with a specific tip

- All text associated with sample or configuration files is presented in special font, as in

    ```
    # This is a sample configuration file for…
    # #################################################
    [options]
    recursion=true
    silent=true
    checkonly=false
    ```

- Differences depending on the revision level of RFTerm installed are presented as

    > ⚠️ The Cursor tab is now part of the Session configuration. It is located at **Session** | **Configure** | **Cursor** tab. Please refer to Chapter 4, "Session Configuration" for cursor configuration options.

Note: This convention is generally used when a parameter choice has been moved to a different RFTerm screen. It does not document every parameter or tab added to RFTerm.   Please consult your LXE representative with any questions regarding RFTerm revisions.

Getting Help

LXE user guides are now available on one CD and they can also be viewed/downloaded from the LXE ServicePass website. Contact your LXE representative to obtain the LXE Manuals CD.

Note: Obsolete/archived equipment and software manuals are not available on the LXE Manuals CD. They are available for download from the LXE ServicePass website only.

You can also get help from LXE by calling the telephone numbers listed on the LXE Manuals CD, in the file titled "Contacting LXE". This information is also available on the LXE website.

Explanations of terms and acronyms used in this guide are located in the file titled "LXE Technical Glossary" on the LXE Manuals CD and the LXE ServicePass website.

Feature Highlights

Multiple Host Sessions

RFTerm supports up to four simultaneous pre-defined host sessions. All sessions run under a single application process. Interactive, per session configuration settings are maintained. Users may connect with any permutation of 3270, 5250 or VT host sessions. HotKeys and menu context are available to jump between these sessions.

HotSpots

A HotSpot is an invisible field on the mobile client screen where a user can tap with a stylus to execute a function. A HotSpot allows a user to interact with the host application with minimal needs for the special key pads. Instead the user can directly tap on the text in the display to invoke the desired operation.

Screen Panning

Panning allows an additional row and column on the display by providing an alternative to horizontal and vertical scroll bars. Imagine the host application screen as a large sheet under the display on the handheld device. Screen Panning allows users to "tap" and "drag" this sheet in any direction to move hidden areas of the host display into view on the handheld device without the use of scroll bars.

Configurable KeyBar

The configurable KeyBar allows users to customize a set of soft buttons to control and invoke any host specific keys or emulator operation. Users can select from a set of pre-defined KeyBar templates. In addition they can also configure up to four custom templates of their own. Users can navigate between a selected set of templates.

SmartPads

All host applications have specific keyboard requirements. The SmartPad is a floating button pad which provides support for special emulation host keys. Depending upon the current configured host emulation type, the appropriate SmartPad for that emulation is displayed. The SmartPad supports all the popular 3270, 5250 and VT host keys.

Note: This option is not available on Microsoft® Windows XP or Windows 2000 devices.

Access Control

Access Control allows administrators to prevent users from changing the TE configuration for deployed devices. Administrators can require a password to change the session configuration and application settings.

Device Lockdown

Lockdown allows administrators to prevent users from exiting the TE client. Lockdown also prevents users from running any other application on the device.

Configurable Fonts

This option allows users to increase or decrease the font sizes of displayed text.

Color Schemes

Support for color displays. A color scheme is a collection of colors mapped to a set of screen display attributes. Users can select from pre-defined color schemes or create a custom scheme based upon their preferences. Applications dynamically detect LCD displays to self configure for a two color scheme (Black on White).

Automatic Login

Users can configure one or more host sessions to automatically launch when the TE application is started. Pre-defined login scripts can be configured to launch automatically and log into the host system.

Automatic Session Persistence

Latest values of session configuration are automatically saved for every session.

Dynamic Cursor View Modes

Multiple cursor modes are available to support automatic scrolling, so that the cursor / input field is visible. These view modes are configurable on a per client session basis. The screen display window will "track" and follow the cursor.

Text InputTool

The Text InputTool improves upon the typical Windows CE .NET soft keypad by sending complete text strings to the display window at the current cursor location. A user can cache often-used text strings in the Input Tool.

Note: This option is not available on Microsoft Windows XP or Windows 2000 devices.

Device / LU Name Support

For IBM emulations (3270 and 5250), logical unit device name resource configuration is supported. This is used during connection negotiations.

Automatic Re-connection

If for some reason the network connection to the host is lost, the user can configure the session to automatically attempt to re-connect to the host.

Full VT220 Support

This includes support for multinational character sets and NRC support.

VT Line Mode

This is a special optimization for VT host sessions. Data is buffered locally within the device prior to sending it to the host. Line mode optimizes usage of the network bandwidth and host resources.

VT Answerback String

This is a configurable text string that certain host applications require to identify the mobile client.

HotKeys

Predefined HotKeys, shortcuts, are available to interact with and invoke application operations.

LXE Block Mode

Block mode communication is a communication mode that enables the device to send an entire block of data to the host at one time. Block mode greatly reduces RF traffic by sending data in this way. Block mode minimizes RF overhead better than the other modes. Thus, block mode supports more users on an RF channel and allows faster response time.

Local echo is a process that allows each character to display locally at the device as the user presses the keys. The host does not see these characters until the user presses the [Enter] key.

Note: *In most cases, operations using programming tools that include screen painters cannot use block mode communication. Screen painters or designers usually come with programming tools such as Oracle, Ingress, Informix, and Powerhouse, which normally use character mode communications.*

Programmable Function Keys

User can configure all available function keys, and the 4 arrow keys, to perform text and control character inputs. Control characters and characters not accessible from the keyboard can be represented using the hat encoding table in Appendix B.

Printing

User can print the screen for any session using the shortcut key [Ctrl] [Alt] [P], or by sending print commands to the client from the host in 5250 and ANSI sessions.

Help

User can display frequently used RFTerm shortcut keys and LXE contact information by pressing [Ctrl] [H].

Double Byte Support

RFTerm supports Simplified Chinese, Traditional Chinese, Korean and Japanese font sets for VT emulations based upon TELNET data being received in standard DBCS format.

Installation Notes

Installation

Your device may require special installation procedures. See Appendix C "Installation" for details.

Upgrading

When you upgrade RFTerm to a new release version, it will be necessary to perform the steps outlined in Appendix C "Installation".

About RFTerm

Information about RFTerm can be viewed by selecting **File | About**

Figure 1-1 About RFTerm

The **About** sceen contains important information about RFTerm, including:

- The platform on which RFTerm is installed (Windows XP, Windows CE.net, etc)
- The revision level of RFTerm

Click to **OK** button to dismiss the **About** screen.

Chapter 2 Quick Start

RFTerm Screens on Different Sized Platforms

The RFTerm screen displays are designed to function on most Windows-based computers, regardless of the physical screen size.

For example, a full size screen on a desktop computer monitor, a laptop computer monitor, mobile computers that have a half screen (approximately 8" wide by 3" high), others with one-quarter screens and yet more devices with screens that are even smaller in physical display size.

RFTerm menus and editing parameters, save and exit options are available on-screen, where needed, regardless of the size of the screen – on-screen they may look or be named differently from one device to the other yet they will perform the same function when activated.

This guide makes the assumption that the user or systems administrator is familiar with Microsoft Windows standard navigation/performance for each of their devices running RFTerm.

Therefore, the sections that follow describe only those Windows capabilities that are unique to RFTerm. Screen displays in this document reflect those seen on devices with a Windows XP operating system.

For specific Windows commands or navigation instruction, please refer to Windows on-line help or commercially available Windows operating system reference manuals.

Figure 2-1 RFTerm Splash Screen on the HX2

Figure 2-2 RFTerm Splash Screen on the MX5, MX6, MX7 and MX8
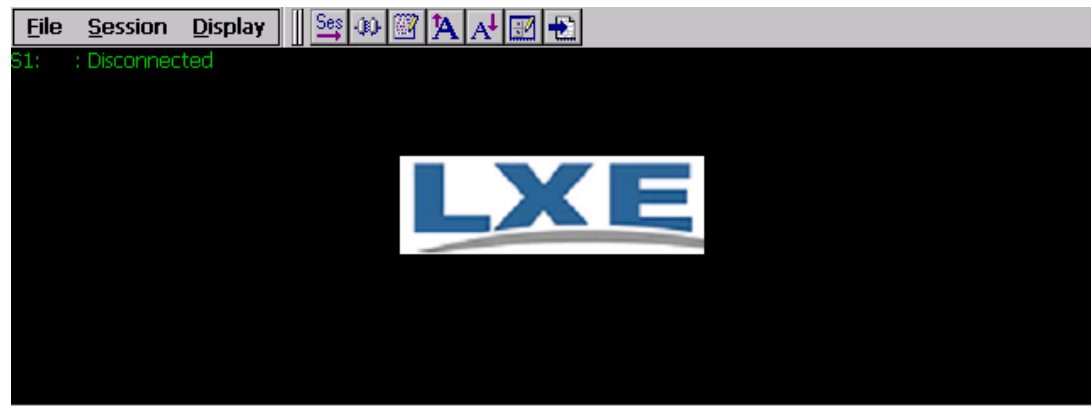
Figure 2-3 RFTerm Splash Screen on the MX3X, VX3X and VX6
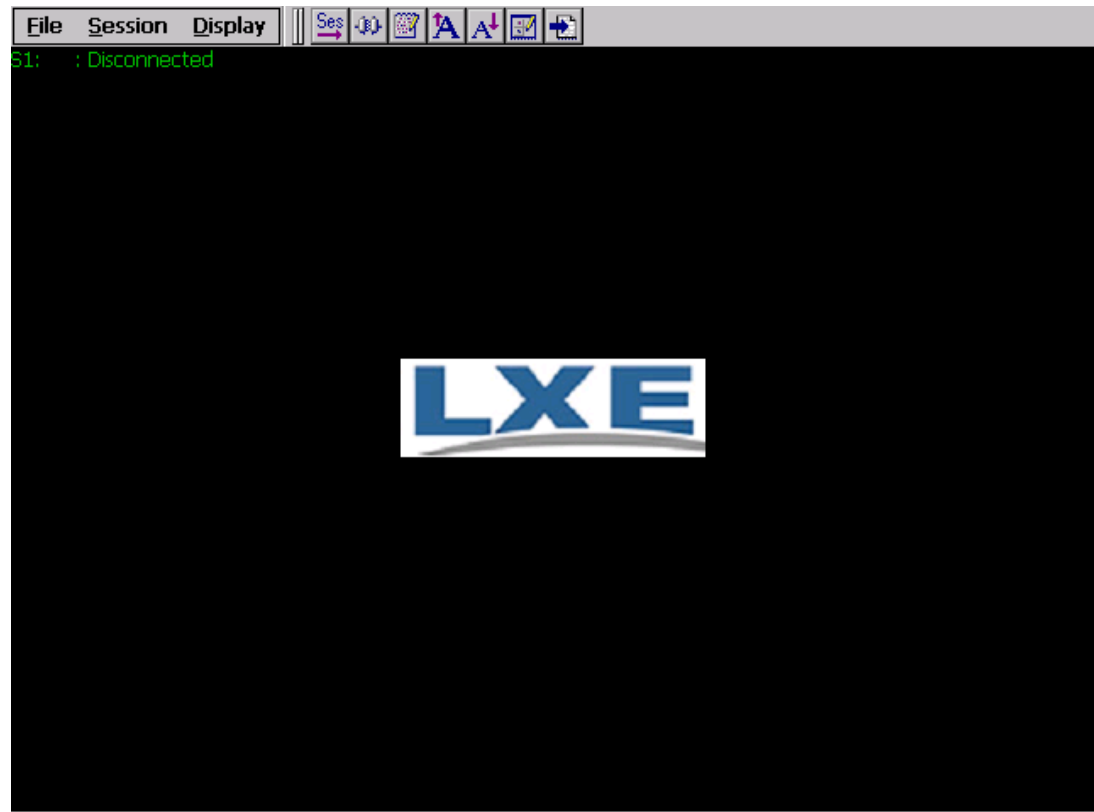
Figure 2-4 RFTerm Splash Screen on VX5 and VX7

Quick Start

This section is for advanced users who can comfortably navigate within Windows and Windows CE and are familiar with host terminal emulation details. Follow these steps to connect to your host application with minimal configuration setup. For details on various configuration parameters, it is recommended that you read through the following sections.

1. Make sure the mobile client network settings are configured and functional. If you are connecting over wireless LAN (802.11b/g), make sure your mobile client is communicating with the Access Point.

2. If the RFTerm splash screen does not automatically appear after startup or rebooting, go to **Start | Programs**, and select **LXE RFTerm** or double-click the **RFTerm** icon on the Windows desktop.

3. Select **Session | Configure** from the application menu and select the **Host Type** that you require. This will depend on the type of host system that you are trying to connect to; i.e. 3270 mainframe, AS/400 5250 server or VT host.

4. Enter the **Host Address** of the host system that you wish to connect to. This may either be a DNS name or an IP address of the host system.

5. Update the Telnet **Port** number, if your host application is configured to listen on a specific port. If not, just use the default Telnet port.

6. Select **OK.**

7. Select **Session | Connect** from the application menu or tap the **Connect** button on the Command Bar. Upon a successful connection, you should see the host application screen displayed.

8. During the connection process, the following popup dialog will appear:
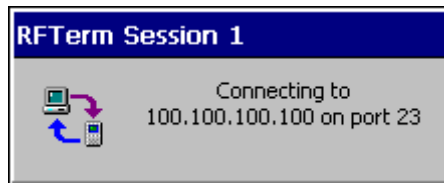
Figure 2-5 Connection in Progress

Quick Tips

Screen Panning

- For all connected sessions (3270, 5250 and VT), you can "tap-and-drag" the display screen in any direction with your stylus. Just make sure that HotSpots are enabled in your File | Settings | Options configuration. Panning stops once the edge of the host screen has reached the display border.

HotSpots

- For all connected sessions (3270, 5250 and VT), you can "double tap" anywhere on the screen to invoke an Enter (or Send) operation.

- To use HotSpots, you can directly tap the function key text string on the display rather than use a SmartPad. Refer to the "HotSpots" section for details.

Predefined HotKeys (Shortcuts)

- Almost every application operation has a HotKey associated with it. HotKeys provide an efficient alternative to invoke an operation or action. Refer to the "HotKey Shortcuts" table in Chapter 5.

Readability

- For more screen real estate, you can minimize the application menu. You can also move open windows to anywhere on the screen (unless the windows are locked to prevent edit).

- You can select different background, foreground and attribute colors for enhanced visual readability. For mobile clients with LCD display screens, the default color scheme is automatically detected and configured to "Bright White" for optimal readability.

Data Input

- If you have the same data strings that you input all the time, you can cache it in the data input tool. You can then "send" the text string to the emulator with a single stylus tap. Refer to the "Text InputTool" section for details.

  Note: This option is not available on Windows XP or Windows 2000 devices.

Configurable KeyBar

- The soft KeyBar can be configured to perform most emulator and host operations with a single stylus tap. Refer to the Configurable KeyBar section for details.

Access Control

- You can prevent a user from changing the TE configuration by specifying an access password. Refer to the access control section for details.

Application ToolBar

- An integrated ToolBar is provided for touch screen navigation.

- Please refer to "Appendix A – Key Maps", section titled "Icon / Button Identification" to identify the function for the icons that are included in the different application touch KeyBars.

Miscellaneous

- The RFTerm emulator will automatically save configuration settings for the four most recent sessions, S1 – S4 as displayed in the menu. The configuration settings are saved in the mobile client registry. Please refer to the mobile client reference guide for instruction.

- To launch RFTerm automatically, please refer to the mobile client reference guide.

- Support tools and procedures are available to provide deployment configurations to self-install and restore application and mobile client configurations during a cold boot.

Chapter 3 RFTerm Settings

Introduction

This section describes application wide settings. All these parameters may be set using the application settings dialogs accessed from the **File | Settings** application menu. Parameters configured through the **File | Settings** dialog apply to all terminal emulation sessions.

- All shortcut keys can be enabled or disabled at once. Shortcut keys can be disabled or enabled individually as well.
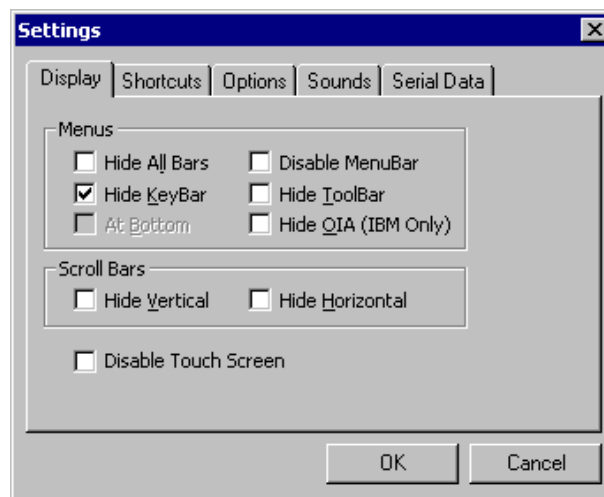
Figure 3-1 Settings Tabs

⚠	The Cursor tab is now part of the Session configuration. It is located at **Session** | **Configure** | **Cursor** tab. Please refer to Chapter 4, "Session Configuration" for cursor configuration options.

Display

The client screen display may be configured to optimize screen available area and readability. These options may be configured for connected or non-connected host sessions. HotKeys are available to toggle the options on and off.

Figure 3-2 Settings – Display Tab

- From the application menu, select **File | Settings**.
- A **Settings** dialog box will come up.
- Choose the **Display** tab.

Parameters

Menus

ToolBars, KeyBars, Shortcut Keys and the MX6

By default, the MX6 ToolBar button function is configured as part of the KeyBar on the MX6 so the checkbox for the ToolBar does not exist on the Settings | Display tab.

MX6 ToolBar buttons can be removed from the KeyBar.

MX6 ToolBar buttons can be hidden by using **[Ctrl][Shift][K]** which makes the entire KeyBar hidden on the MX6.

The Hide ToolBar shortcut key **[Ctrl][Shift][T]** is not available on the MX6.

Hide All Bars

Checking this box will hide the file menu, the KeyBar and the ToolBar. Pressing **[Ctrl][Shift][L]** will alternate between the hidden and visible states. This will override each individual item listed below. Not available on the MX6.

Disable/Hide MenuBar

Checking this box will hide the MenuBar on Windows CE devices. Pressing **[Ctrl][Shift][M]** will alternate between the hidden and visible states.

When the MenuBar is hidden, the KeyBar is displayed on the bottom row, where the MenuBar is normally shown. When MenuBar is toggled on, the KeyBar is displayed on the row above the MenuBar.

Note: This parameter is titled Disable MenuBar on Windows XP devices and Hide MenuBar on CE devices.

Hide KeyBar

Checking this box will hide the KeyBar. The default is checked (hide KeyBar). Click the K icon on the MenuBar to launch the KeyBar. Pressing **[Ctrl][Shift][K]** will alternate between the hidden and visible states.

When the MenuBar is hidden, the KeyBar is displayed on the bottom row, where the MenuBar is normally shown. When MenuBar is toggled on, the KeyBar is displayed on the row above the MenuBar.

MX6 ToolBar buttons can be hidden by using **[Ctrl][Shift][K]** and the entire KeyBar is hidden on the MX6.

Hide ToolBar

Checking this box will hide the ToolBar. Pressing **[Ctrl][Shift][T]** will alternate between the hidden and visible states. Not available on the MX6.

(Menu) at Bottom

Checking this box will move the menu to the bottom of the screen. Pressing **[Ctrl][Shift][B]** will alternate the location between the top and bottom of the screen. The Menu at Bottom option is disabled on the MX6. The menu will always display at the bottom of the MX6 screen.

Note: This option is not available on Windows XP or Windows 2000 devices.

Hide OIA (IBM only)

Checking this box will toggle hiding of the OIA or the Operator Information Area. This option applies only to IBM 3270 and 5250 emulations.

Pressing **[Ctrl][Shift][O]** or **[Alt] [T]** alternates between the hidden and visible states.

Scroll Bars

Hide Vertical Scroll Bar

Checking this box will hide the vertical scroll bar. Pressing **[Ctrl][Shift][V]** will alternate between the hide and visible states.

Hide Horizontal Scroll Bar

Checking this box will hide the horizontal scroll bar. Pressing **[Ctrl][Shift][H]** will alternate between the hide and visible states.

Disable Touch Screen

When checked, the touchscreen is disabled while still allowing RFTerm Hotspots to function using touch. The default value for Disable Touch Screen is "Not Disabled" (blank).

MX6 35-key Buffering

A **35-key buffering checkbox** is available on the **Settings** | **Display** tab on the MX6.

If the MX6 transmits keypresses before the desired key sequence is completed, (for example, a lengthy or complex login/password combination on a 35-key MX6) enable the 35-key buffering checkbox and RFTerm inserts a 500ms delay between same key keypresses.

If it becomes necessary to adjust the 500ms delay, please contact LXE Customer Support for assistance.

Shortcuts

Keypress shortcuts can be enabled or disabled from the "Shortcuts" tab.
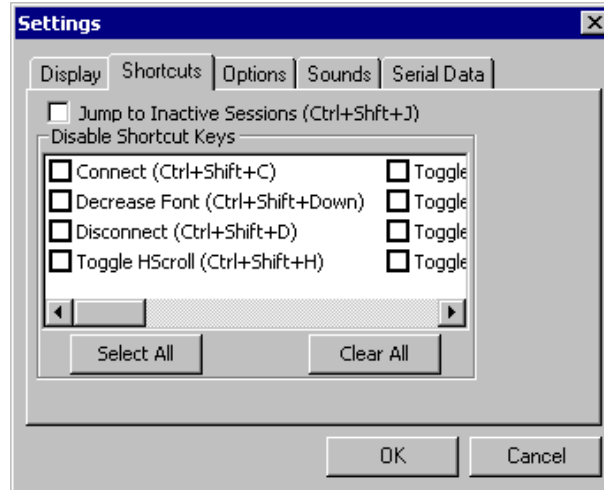
Figure 3-3 Settings – Shortcuts Tab

- From the application menu, select **File | Settings**.
- A **Settings** dialog box will come up.
- Choose the **Shortcuts** tab

Parameters

Jump to Inactive Sessions

When the Jump to Inactive Sessions item is selected, the user can use the shortcut key sequence **[Ctrl] [Shift] [J]** to jump to an inactive session that has a host name specified. The inactive session remains dormant when disconnected, but when it becomes the active session again, it automatically attempts to reconnect and run the Autologin script.

Disable Shortcut Keys

When the Disable Shortcut Keys item is checked, RFTerm prevents the user from accessing all shortcut keys *with the exception of Session, Configure and Application Exit shortcut keys*. This also activates the "Allow Session Shortcuts" check box. F1, F2, F3, and F4 function keys remain active. See Appendix A for a complete list of RFTerm shortcut keys.

Buttons

Select All   Click this button to disable all shortcut keys, in the list, at once. Click a shortcut key checkbox to enable the individual shortcut while the rest remain disabled (checked). The Select All button has no effect on the Jump to Inactive Sessions checkbox.

Clear All   Click this button to enable all shortcut keys, in the list, at once. Click a shortcut key checkbox to disable (check) the individual shortcut while the rest remain

enabled. The Clear All button has no effect on the Jump to Inactive Sessions checkbox.

Shortcuts Active Until Disabled

The shortcut is disabled when the checkbox has a check mark. See Also: "HotKey Shortcuts" in Chapter 5 for a complete list of default Hotkeys.

Note: Depending on mobile client screen size constraints, the parameter name may be abbreviated based on the parameter names shown in the list that follows.

Parameter	Shortcut Key	Description
Configure	[Ctrl][Shift][N]	Configure a host connection
Settings	[Ctrl][Shift][E]	Configure application settings.
Connect	[Ctrl][Shift][C]	Connect to the host *(Manual Reconnect shortcut)*
Decrease Font	[Ctrl][Shift][Down Arrow]	Decrease font size
Disconnect	[Ctrl][Shift][D]	Disconnect a connected host session *(Manual Reconnect shortcut)*
Toggle HScroll	[Ctrl][Shift][H]	Display or hide horizontal scroll bar
Toggle SmartPad	[Ctrl][Shift][P]**	Display or hide SmartPad
Toggle InputPad	[Ctrl][Shift][I]**	Display or hide Text InputTool
Toggle All Bars	[Ctrl][Shift][L]	Display or Hide all bars (Menu, Tool, Key)
Toggle MenuBar	[Ctrl][Shift][M]**	Display or hide MenuBar
Toggle ToolBar	[Ctrl][Shift][T]	Display or Hide ToolBar
Toggle KeyBar	[Ctrl][Shift][K]	Display or Hide KeyBar
Toggle VScroll	[Ctrl][Shift][V]	Display or Hide vertical scroll bar
Move All Bars	[Ctrl][Shift][B]**	Display all bars at the bottom
Exit RFTerm	[Ctrl][Shift][X]	Exit the emulator
Toggle OIA Bar	[Ctrl][Shift][O] or [Alt][T]	Hide and unhide status bar (OIA)
Increase Font	[Ctrl][Shift][Up Arrow]	Increase font size
Session Save	[Ctrl][Shift][S]	Invoke the session save dialog

Parameter	Shortcut Key	Description
Next Session	[Ctrl][Shift][J]	Jump to next <u>connected</u> host *(Session Control shortcut)*
Scroll Down	[Shift][Down Arrow]	Scroll down one row
Scroll Left	[Shift][Left Arrow]	Scroll left one column
Scroll Right	[Shift][Right Arrow]	Scroll right one column
Scroll Up	[Shift][Up Arrow]	Scroll up one row
Advance Down	[Ctrl][Down Arrow]	Scroll down one row segment (*see Note*)
Advance Left	[Ctrl][Left Arrow]	Scroll left one column segment (*see Note*)
Advance Right	[Ctrl][Right Arrow]	Scroll right one column segment (*see Note*)
Advance Up	[Ctrl][Up Arrow]	Scroll up one row segment (*see Note*)
Toggle Bold	[Ctrl][Shift][1]	Toggle between bold and normal font
Print Screen	[Ctrl][Alt][P]	Print the screen of the active session
Shortcut Help	[Ctrl][H]	Display frequently used shortcuts

** Not supported on Windows XP / 2000 devices.

Note: Depending on mobile client screen and font size constraints, the segment area used by an Advance Down, Advance Left, Advance Right and Advance Up key sequence is approximately one-quarter of the total area available.

Options

General options may be configured and updated from the "Options" tab.
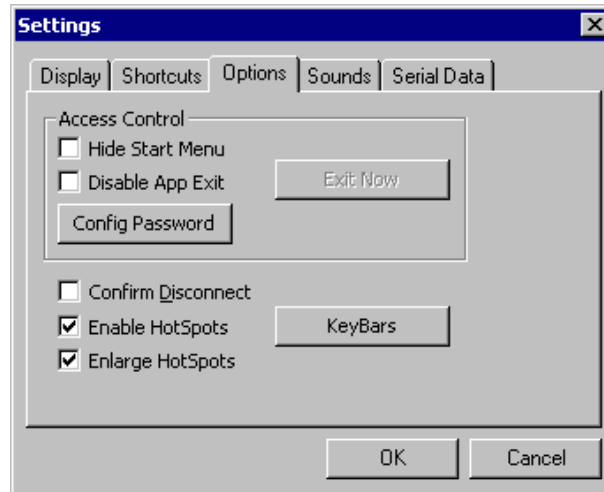
Figure 3-4 Settings – Options Tab

- From the application menu, select **File | Settings**.
- A **Settings** dialog box will come up.
- Choose the **Options** tab

Parameters

Access Control

Hide Start Menu

When the Hide Start Menu item is selected, RFTerm prevents the user from accessing functionalities within the task bar.

Windows XP/2000

This feature makes the application full-screen and also disables window resizing.

MX6

When the Hide Start Menu / Disable Start Menu item is selected, neither the Start Menu nor the OK button respond to a stylus tap. Use an Enter keypress to close the Settings and Configure window.

Disable App Exit

When enabled, this option disables the application exit button and the [Alt][X] shortcut key. This prevents the user from exiting the application.

Exit Now

This button is enabled only when the "Disable App Exit" option is checked. It allows the administrator to save the configuration and exit when "Disable App Exit" is checked. Typical use would be for an administrator to set all configuration settings including a config password and exit the application using this button. Subsequently, users will not be able to exit the application and the config password will be required to change the configuration.

Config Password

The Config Password button prompts the user for a dialog access password. When a password is set, users must enter this password prior to viewing and modifying the session and settings configuration. This capability enables administrators to prevent users from changing the application configuration settings in controlled environments.

Confirm Disconnect

When enabled, the user is prompted before every attempt to disconnect an active session.

Enable HotSpots

Checking this box will enable HotSpots and Screen Panning on mobile clients with touch screens. A HotSpot is a field on the screen where a user can tap with a stylus to execute a function. This allows a user to interact with a host application with minimal need for special keys. Checking this box is a requirement for the Enlarge Hotspots parameter.

For example

A simple example might be the use of F Keys. An operation associated with an F key might be displayed on the screen as "F1 = Help". RFTerm automatically detects this as a HotSpot and will simulate an F1 key being pressed on the keyboard when you tap on the F1 text on the screen display. Refer to the HotSpots section for further details.

Note: Reprogramming the function keys using the Fn Keys tab under the Configure dialog will alter the operation of a function key HotSpot.

Enlarge HotSpots

Prerequisite: Enable HotSpots must be enabled (checked).

Checking this box increases the size of the hotspot area, slightly, radiating from the center of the hotspot. If a screen does not have a hotspot area, this parameter setting is ignored until a screen is presented that has a hotspot area. For example, enlarging the hotspot area may assist clicking the correct hotspot even if the mobile client is being jostled by outside influences.

When Enable HotSpots is disabled, the Enlarge HotSpots parameter is dimmed and cannot be activated.

KeyBars

This button is used to configure the KeyBar. KeyBar configuration consists of both the KeyBar Cycle and any Custom KeyBars. The KeyBar function keys can be reprogrammed using the Fn Keys tab under the Configure dialog.

> ⚠ The **KeyBars** button was previously labeled **Config Keys**.

KeyBar Cycle

A KeyBar is the set of keys visible at one time. The KeyBar Cycle is a collection of KeyBars that can be visible. The user "cycles" through the collection of KeyBars by tapping the arrow buttons on either end of the current KeyBar. Each key on the KeyBar is associated with a host or emulator operation. Users are allowed a maximum of eighteen KeyBars, four of which can be customized. The same KeyBar may be added multiple times to the KeyBar Cycle.
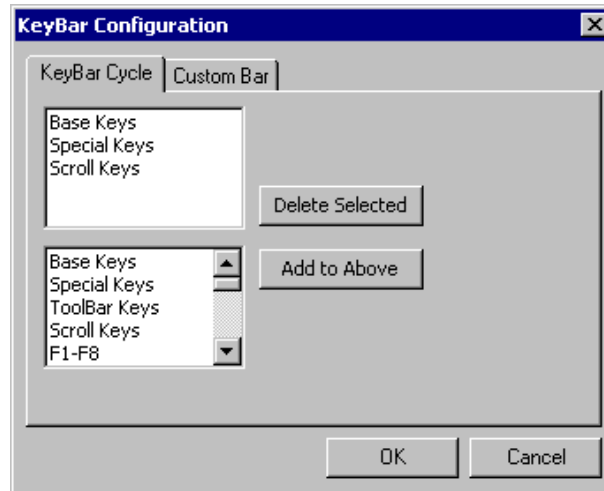
Figure 3-5 KeyBar Configuration

Custom Bar

A Custom KeyBar ("Custom Keys 1" through "Custom Keys 4") can also be included in the KeyBar Cycle. Tapping the "Custom Bar" tab allows the user to configure a custom KeyBar. There are over 180 buttons to choose from, see the Appendix A, "Key Maps" for all available choices.
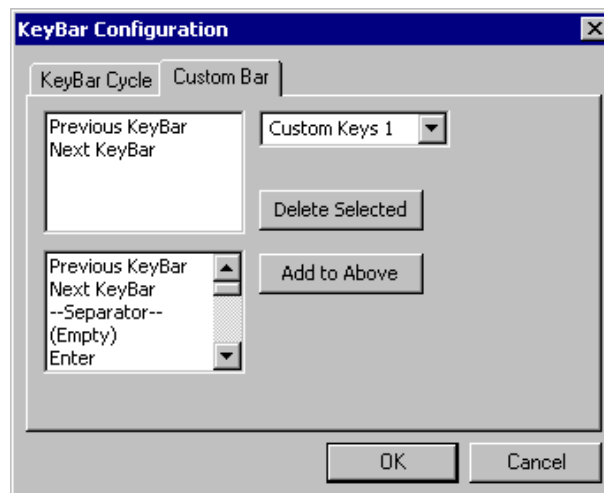
Figure 3-6 KeyBar - Custom Bar

To edit a Custom KeyBar

- In the KeyBar Cycle tab, add a Custom Bar to the KeyBar Cycle

- In the Custom Bar tab, select the Custom Bar that you want to configure

- Select the Key or operation that you want to appear on this custom bar

- Tap "Add to Above". The selected key / operation will appear in the list.

- Delete unwanted keys by selecting them and tapping "Delete".

You can add up to a maximum of 8 key buttons on each Custom Bar (fewer on some devices). The entry for "Previous KeyBar" cannot be removed and must exist in each KeyBar to allow for "cycling" between KeyBars. Typically, the last key should be "Next KeyBar" for cycling to the next KeyBar.

One powerful capability of the KeyBar is the ability to associate keys with the Text InputTool. Key entries "Input 1" through "Input 20" are tied to the corresponding twenty entries in the Text InputTool. Thus if you have a text string configured in the data input tool, this string can be submitted to the host application by tapping on the "Input X" key in a Custom KeyBar. Tapping this key will send the complete text string to the current cursor location. For VT client sessions, escape sequences can be added to the Text InputTool text. This allows users to configure custom escape sequences as required by their host applications.

Escape sequences can be entered into the Text InputTool in the following format:

\e = Escape
\n = Newline
\r = Enter or Return
\t = Tab
\xDD = Hexadecimal value

Notes:

- The list of keys that are available for a Custom KeyBar includes "Escape". The Escape icon is identical for VT Escape and 3270 Escape, both listed in Appendix A "Key Maps", section titled "RFTerm ToolBar" icon listing. When setting up a Custom KeyBar for VT or IBM, and VT Escape or 3270 Escape is required, select the "Escape" icon.

Sounds

The Alarm on Match Text only sounds when a form is sent to RFTerm from the host. Entering data or moving the cursor around on the screen does not cause extra beeps when the alarm text remains unchanged. When the user exits and then returns to the Alarm on Match Text menu, another beep will occur.

Note: Before selecting a user-installed sound or WAV file to be played on a mobile client, the appropriate WAV file or sound file must be available in the mobile device operating system persistent folder on the storage card. This will ensure the WAV file is saved instead of discarded when the device is rebooted.

If there is no WAV file in the folder, the text boxes are blank. When there is a WAV file in the folder, the Good Beep/Bad Beep and Alarm Sound drop down boxes are preset with the default value <NONE> until changed.
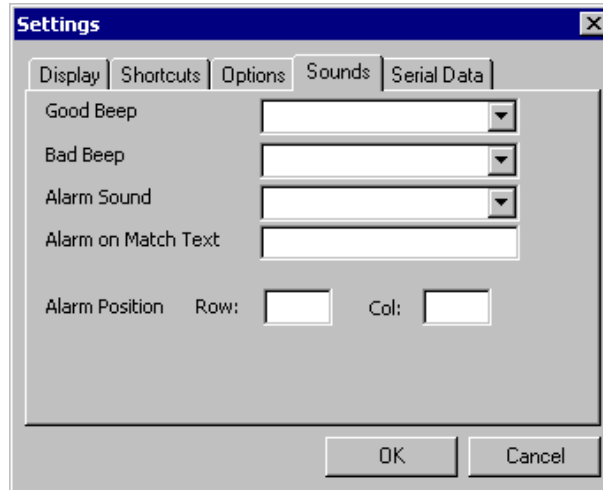
Figure 3-7 Settings – Sounds Tab

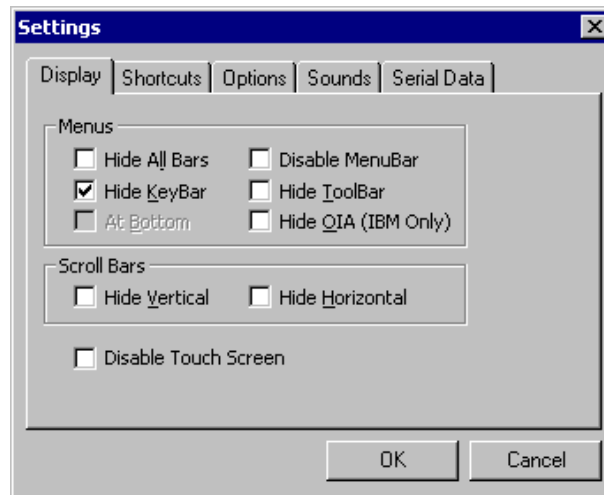- From the application menu, select **File | Settings**.
- A **Settings** dialog box will be displayed.
- Choose the **Sounds** tab.

Parameters

Good Beep Wav

Select the sound from the drop-down box to play when a good scan occurs. The default is None.

Note: Before enabling the Good Beep Wav, verify that "Good Scan" sounds are disabled on the mobile client using the device's control panel applet (e.g., Volume and Sounds Properties and/or Scanner Properties). When Good Scan sounds are enabled in RFTerm and on the mobile client, there may be double sounds emitted by the mobile client during a "Good Beep" function.

Bad Beep Wav

Select the sound from the drop-down box to play when a bad scan occurs. The default is None.

Note: Before enabling the Bad Beep Wav, verify that "Bad Scan" sounds are disabled on the mobile client using the client's Control Panel applet (e.g., Volume and Sounds Properties and/or Scanner Properties). When Bad Scan sounds are enabled in RFTerm and on the mobile client, there may be double sounds emitted by the mobile client during a "Bad Beep" function.

Beep on Error

When the Windows System Sound (Exclamation Point) is desired as the Bad Beep sound, the user must change the WAV file associated with the Operating System on the mobile client (e.g. "Volume and Sounds"). See the mobile client specific Reference Guide.

Alarm Sound Wav

Select the sound to play when text shown in "Alarm on Match Text" is displayed. The default is None.

Note: Before enabling the Alarm Sound Wav, verify that all sounds are disabled on the mobile client using the client's Control Panel applet (e.g., Volume and Sounds Properties). When Alarm Sounds are enabled in RFTerm and on the mobile client, there may be double sounds emitted by the mobile client during an RFTerm Alarm function.

Alarm on Match Text

Enter the text to be displayed on the user's screen (for example, "Login") when a response from the user is desired. The default is blank.

After the "Alarm Sound WAV" and "Alarm on Match Text" run, RFTerm then waits for a user response.

Alarm Position

Enter the row and column starting position for the Alarm on Match Text to be displayed on the user's screen. The default is Row=0 Column=0.

The alarm sounds when it discovers the Alarm on Match Text at the specified row and column. The alarm does not sound when the Alarm on Match Text is located at any other position.

Serial Data

Note: The serial ports (for input) are only configurable on the VX5.
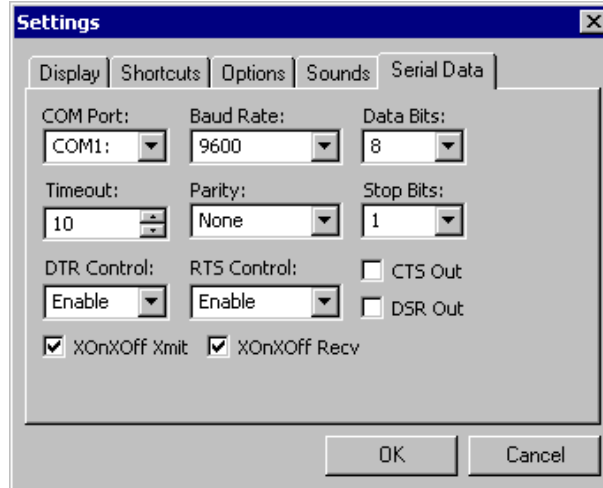
Figure 3-8 Settings – Serial Data Tab

- From the application menu, select **File | Settings**.
- A **Settings** dialog box will be displayed.
- Choose the **Serial Data** tab.

Should RFTerm be unable to apply any of the COM Port settings, the following popup dialog will appear.
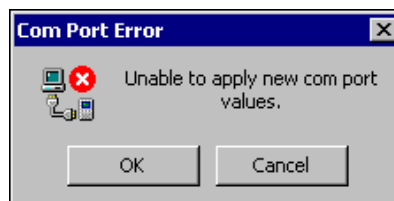
Figure 3-9 COM Port Error

Parameters

COM Port

COM ports available are dependent on the configuration of the mobile device. Select the COM port from the drop down menu. Options may include: COM1, COM2, None, etc.

Baud Rate

Specifies the baud rate at which the COM port operates.

Data Bits

Specifies the number of data bits to be used. It is one of the following values: 7, or 8.

Timeout

Specifies port timeout in seconds.

Parity

Specifies the parity scheme to be used. It is one of the following values: even, mark, none, odd and space

Stop Bits

Specifies the number of stop bits to be used. It is one of the following values: 1, or 2.

DTR Control

Specifies the data-terminal-ready flow control. The options are enable, disable and handshake.

RTS Control

Specifies the request-to-send flow control. The options are enable, disable, handshake and toggle.

CTS Out

Specifies if the CTS (clear-to-send) signal is monitored for output flow control.

DSR Out

Specifies if the DSR (data-set-ready) signal is monitored for output flow control.

XonXoff Xmit

Specifies if XON/XOFF flow control is used during transmission.

XonXoff Recv

Specifies if XON/XOFF flow control is used during reception.

Stored Forms

A stored form is a combination of valid ANSI/DEC/LXE commands and text that, when executed by RFTerm and displayed on the screen, appears as a fill-in form.

RFTerm places the commands and text that comprise the form in a device-dependent folder:

- **Windows**/LXE/Stored Forms folder (VX5) or

- **System**/LXE/RFTerm/Stored Forms folder (CE devices) or

- **IPSM/**LXE/RFTerm/Stored Forms folder (MX6).

Creating Stored Forms

Stored forms are written by the host application programmer and sent from the host to the mobile client. You cannot create a form from the mobile client.

The maximum number of stored forms supported by RFTerm is 100 and the maximum size of each stored form is 2K. The /Stored Forms folder is not automatically cleared upon a mobile client cold boot.

See also: Chapter 6, "ANSI" and Appendix B, "ANSI Quick Reference Guide", section titled "Forms" for command codes for stored forms.

Chapter 4 Session Configuration

Introduction

This section describes various configuration parameters for setting host connections. All these parameters may be set using the application configuration dialogs accessed from the **Session | Configure** application menu. Except for "Connection" parameters, you can change or update any other parameters at any time, whether the host session is connected or disconnected.

All configuration attributes are associated with the currently active session. Every session can have a different set of configuration attributes.
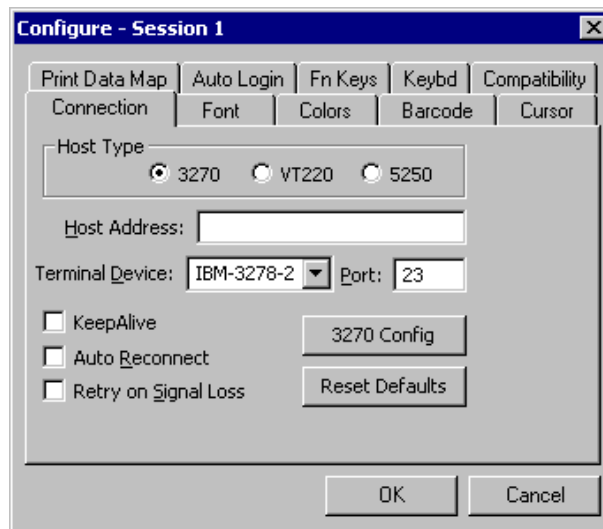
Figure 4-1 Session Tabs

⚠	There are several new tabs available on the **Sessions**	**Configure** screen. To access items added since your software release, please contact your LXE representative for upgrade information.

Connection

Prerequisites for Connection

Before you make a host connection, you will, at a minimum, need to know:

- the alias name or IP address and
- the port number of the host system

to properly set up your host session.

The following figures show the Connection tab of the host session configuration dialog.
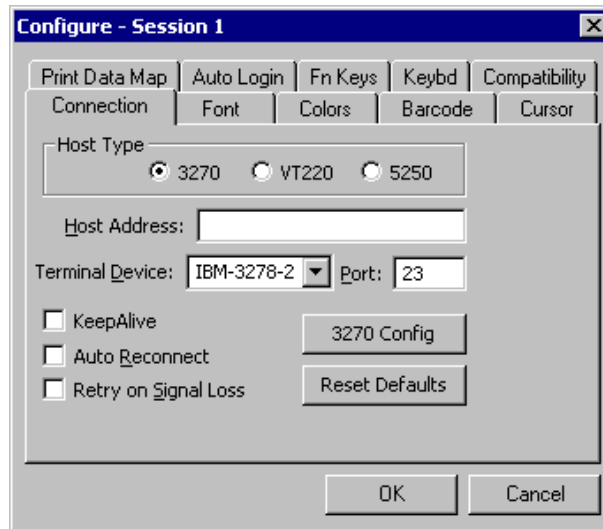
Figure 4-2 Configure 3270 Host Connection
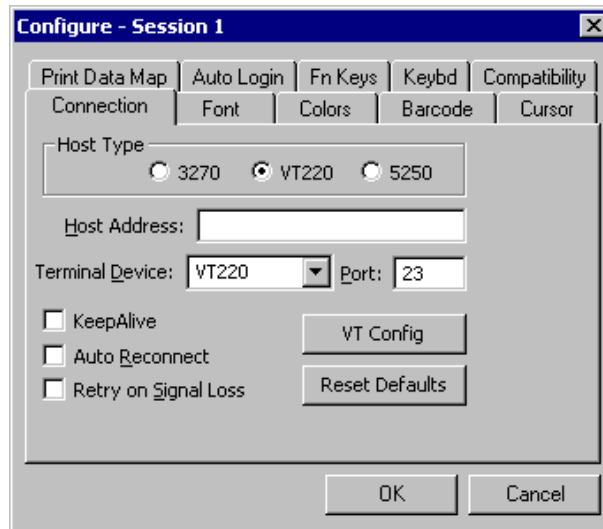
Figure 4-3 Configure VT220 Host Connection
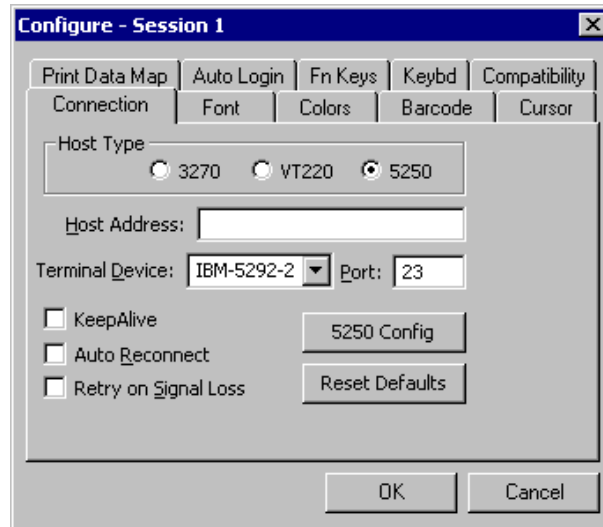
Figure 4-4 Configure 5250 Host Connection

Note: The dialogs dynamically change depending upon the current selection of "Host Type".

1. From the application menu, Select **Session | Configure** or tap the **Configure** button on the ToolBar.

2. A **Configure** dialog box will come up.

3. Choose the **Connection** tab.

Parameters

Host Type

This is the terminal emulation type required by the host. Your choices are 3270, 5250 or VT220. You may select only one emulation type for the session. Other connection options may change depending upon the host emulation type selected.

Host Address

This is the address of a host system or an intermediate gateway managing connections to the host system. Enter either the numeric IP address or the alias name, using up to 64 characters. This option is case-sensitive.
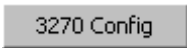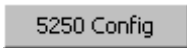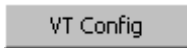
Terminal Device

This is the client type string that determines the specific client to emulate. For 3270 emulations only model 2 screen geometry is supported. VT220 supports VT52, VT100 and VT220 terminal types. Each host emulation type has its own default terminal type.

Telnet Port

This is the TCP socket port number to be used to connect to the host system. The default is 23. The range is between 0 and 65535.

3270, 5250, VT Config

| 3270 Config | 5250 Config | VT Config |

This button, which changes with the Host Type selection, opens a configuration dialog for 3270, 5250 or VT emulation settings.

KeepAlive

Checking the KeepAlive box will send a message to the host every 2 hours (approximately). If the host responds, RFTerm knows the host is still present and will keep the socket open. If the host does not respond to repeated KeepAlive messages, RFTerm will close the connection to the host. The KeepAlive Interval default value is approximately 2 hours.

Auto Reconnect

Checking this box will cause RFTerm to automatically attempt to reconnect to the host should the connection be terminated by the user or the host, other than signal loss.

Retry on Signal Loss

This option enables an automatic reconnection attempt for this session should the radio signal be lost.

VT Configuration

Keys Tab

Figure 4-5 VT Configuration – Keys Tab

- From the application menu, select **Session | Configure** or tap the **Configure** button on the ToolBar.

- A **Configure** dialog box will be displayed.

- Choose the **Connection** tab.

- Select VT220 as the Host Type, press the **VT Config** button.

- Choose the **Keys** tab.

Parameters

Backspace

When in character mode, this option configures the backspace key to send either the Delete character or the Backspace character. The default is Send Delete character.

Enter Key

This option configures the Enter key to send either a carriage return or a carriage return and line feed. The default is Send Carriage Return.

Keyboard

Specifies the international ASCII character set for use with the VT220 emulation. The default is North American.

Screen Tab

Figure 4-6 VT Configuration – Screen Tab

- From the application menu, select **Session | Configure** or tap the **Configure** button on the ToolBar.
- A **Configure** dialog box will be displayed.
- Choose the **Connection** tab.
- Select VT220 as the Host Type, press the **VT Config** button.
- Choose the **Screen** tab.

Parameters

Background

This option controls the background shade for VT host sessions. The default is Dark.

Columns

This option specifies the number of columns to be set for the VT host session to be 80 or 132. The default column value is 80 columns.

See Also: VT Configuration | Answerback Tab | Columns.

Autowrap

Check this option to enable automatic wrapping of text once it reaches the maximum column width. The default is Enabled (checked).

Local Echo

When in character mode, check this option to echo typed text locally on the mobile client.

Options Tab

Figure 4-7 VT Configuration – Options Tab

- From the application menu, select **Session | Configure** or tap the **Configure** button on the ToolBar.
- A **Configure** dialog box will be displayed.
- Choose the **Connection** tab.
- Select VT220 as the Host Type, press the **VT Config** button.
- Choose the **Options** tab.

Parameters

C1 Controls

The C1 Controls option allows the user to select 7 bit or 8 bit ASCII control sequences for the host session. The default is Send 7-bit.

User Preference Supplemental

This is the user preferred supplemental character set. The default is DEC.

Auto Transmit

Checking this box will cause the entire screen to be sent to the host after the user exits the last field on the screen. Before sending the data, the Auto Transmit function evaluates the final cursor movement. If the final cursor movement was caused by a backspace or backtab, the screen is not sent to the host. Note that arrow keys have no effect on Auto Transmit. The default is Disabled (blank).

Send/Receive Mode

The default is Character.

Always Block	RFTerm will remain in Block mode during and after logon. The host can override the current mode.
Always Character	RFTerm will remain in Character mode during and after logon. The host can override the current mode.
Always Line	RFTerm will remain in Line mode during and after logon. The host can override the current mode.
Block	RFTerm starts in character mode until a logon connection occurs, and then it remains in Block mode. The host can override the current mode. A function key press performs an Autosend.
Character	RFTerm starts in character mode until a logon connection occurs, and then it remains in Character mode. The host can override the current mode.
Line	RFTerm starts in character mode until a logon connection occurs, and then it remains in Line mode. The host can override the current mode. A function key press performs an Autosend.

Answerback Tab
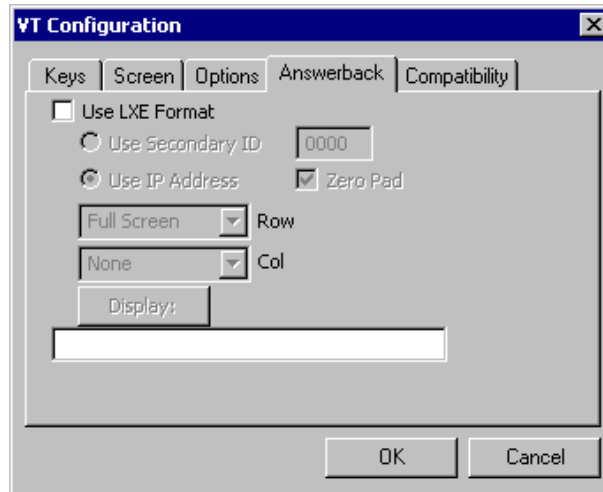
Figure 4-8 VT Configuration – Answerback Tab

- From the application menu, select **Session | Configure** or tap the **Configure** button on the ToolBar.

- A **Configure** dialog box will be displayed.

- Choose the **Connection** tab.

- Select VT220 as the Host Type, press the **VT Config** button.

- Choose the **Answerback** tab.

Parameters

Answerback Message

Host applications may query the VT client for a text message answerback response. This response may be used to identify the client. The answerback string is automatically generated when requested by the host using the ENQ command, or by pressing the Display button. It contains the following items after LXE/q/:

- The number of rows currently visible on the screen.

- The current release number of RFTerm, same as in the About Box.

- The IP address of the device, or an optional secondary ID. The default uses the IP address.

- Optionally, the number of columns currently visible on the screen.

Use LXE Format

When this parameter is enabled (checked), the following options to format the LXE formatted string are active and the Answerback string cannot be edited.

When this parameter is disabled (blank), the user can edit the Answerback string. Other parameters on this dialog box cannot be edited (dimmed). Use IP Address and Zero Pad are Enabled by default.
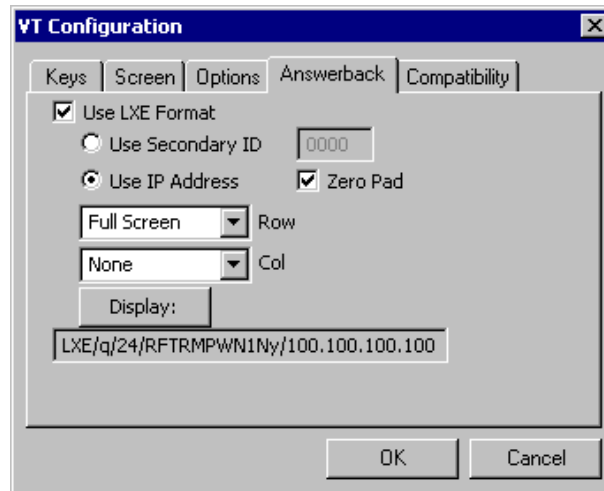
Figure 4-9 Answerback Sample with Use LXE Format

Use Secondary ID

Secondary ID and IP Address are mutually exclusive. Checking one disables the controls for the other. The default Secondary ID is 4 zeros "0000" in alphanumeric format. This is an alternate method to identify the client with a 4-digit number. It is automatically padded with zeros if it contains fewer than 4 characters.

LXE Session Management

To use RFTerm with any of the LXE Session Management products, the Answerback string must be formatted to include a Secondary ID.

Use IP Address

Instructs RFTerm to use the IP Address of the device in the Answerback string. This parameter is disabled when Use LXE Format and Use Secondary ID are enabled. This is the default.

Zero Pad

When Zero Pad is enabled (checked), any of the four numbers in the IP address less than 100 will be padded with leading zeros. For example, with Zero Pad disabled (blank), an IP address is generated as `1.16.253.42`. With Zero Pad enabled, the IP address is generated as `001.016.253.042`.

Row

The default is Full Screen.

None	When selected, sends nothing for the row field in the Answerback string.
Full Screen	When selected, sends 24 for rows in the Answerback string.
Specify	Allows the row information to be set according to the customer's needs. Two characters maximum.

Columns

The default is None.

None	When selected, sends nothing for the column field in the Answerback string.
Full Screen	The setting established in the Columns parameter (80 or 132) on the VT Configuration | Screen tab sets the column size for the Answerback string. See Also: VT Configuration | Screen Tab | Columns.
Specify	Refer to VT Configuration | Screen tab | Columns parameter. Allows the column size, based on the setting in the Columns parameter, to be adjusted according to the customer's needs. Three characters maximum.

Display

The user may press this button at any time to view the answerback string with the selected options.

Note: The string cannot be edited by the user when Use LXE Format is enabled (checked).

Compatibility Tab
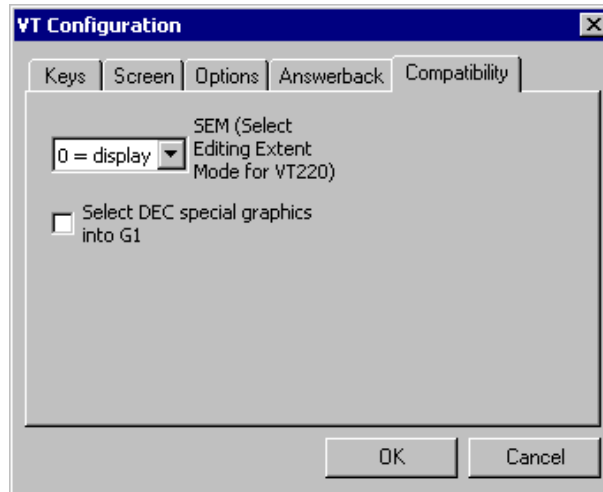
Figure 4-10 VT Configuration – Compatibility Tab

- From the application menu, select **Session | Configure** or tap the **Configure** button on the ToolBar.
- A **Configure** dialog box will be displayed.
- Choose the **Connection** tab.
- Select VT220 as the Host Type, press the **VT Config** button.
- Choose the **Compatibility** tab.

Parameters

SEM (Select Editing Extent Mode for VT220)

The session will use the default SEM assignment unless it is overridden by the host. The default is 0 (display).

Valid values are: 0 = display, 1 = row, 2 = field, 3 = area.

Select DEC Special Graphics into G1

The default is Disabled (unchecked). This option, when enabled, forces the mobile device to be in graphics mode at all times.

5250 Configuration

Options Tab

Figure 4-11 5250 Configuration - Options Tab

- From the application menu, select **Session | Configure** or tap the **Configure** button on the ToolBar.

- A **Configure** dialog box will be displayed.

- Choose the **Connection** tab.

- Select 5250 as the Host Type, press the **5250 Config** button.

- Choose the **Options** tab.

Parameters

Device / LU#

This is an optional IBM parameter. It represents a dedicated LU number or name on the server that you might want to connect through. Default is blank.

Tab = Field Exit

When checked, the Tab key will send a Field Exit command to the host. Default is blank.

Auto Transmit

Checking this box will cause the entire screen to be sent to the host after the user exits the last field on the screen. Before sending the data, the Auto Transmit function evaluates the final cursor movement. If the final cursor movement was caused by a backspace or backtab, the screen is not sent to the host. Note that arrow keys have no effect on Auto Transmit. Default is blank.

Status Line Visible

When Status Line Visible is checked, the status line is displayed on the last viewable row of the mobile client's screen. Default is blank.

Support ANSI Login

Checking this box will enable the 5250 session to perform as a limited VT emulation until an IBM terminal type is negotiated with the host. Default is blank.

Auto Enter

When Auto Enter is checked, an Enter command is sent to the host after a print. Default is blank.

Field Exit on Scan

When Field Exit on Scan: is checked, RFTerm performs a Field Exit automatically if a field is being exited due to a barcode scan. Default is blank.

Compatibility Tab

Figure 4-12 5250 Configuration – Compatibility Tab

- From the application menu, select **Session | Configure** or tap the **Configure** button on the ToolBar.
- A **Configure** dialog box will be displayed.
- Choose the **Connection** tab.
- Select 5250 as the Host Type, press the **5250 Config** button.
- Choose the **Compatibility** tab.

Parameters

Block PF key Inhibit

Enable (check) this checkbox to allow all 5250 PF keys to send. Disable (blank) to block PF key inhibit and data will not be sent when a PF key is pressed.

Block Negative Response for Message Buffer in Use

Enable (check) this checkbox to block a negative response from being sent to the host when the mobile client message buffer is in use at the same time the host sends a command. The default is disabled (blank).

For example, a write to display command is received from the host but the mobile client buffer is being used for a message, so RFTerm sends back a negative response. The host interprets the response as user data.

3270 Configuration

Options Tab

Figure 4-13 3270 Configuration – Options Tab

- From the application menu, select **Session | Configure** or tap the **Configure** button on the ToolBar.

- A **Configure** dialog box will be displayed.

- Choose the **Connection** tab.

- Select 3270 as the Host Type, press the **3270 Config** button.

- Choose the **Options** tab.

Parameters

Device / LU#

This is an optional IBM parameter. It represents a dedicated LU number or name on the server that you might want to connect through. Default is blank.

EAB

This is an optional 3270 terminal emulation parameter. If checked, this will allocate an Enhanced Attribute Buffer. EAB support allows for advanced display features such as color and highlighting. Default is Enabled.

Auto Transmit

Checking this box will cause the entire screen to be sent to the host after the user exits the last field on the screen. Before sending the data, the Auto Transmit function evaluates the final cursor movement. If the final cursor movement was caused by a backspace or backtab, the screen is not sent to the host. Note that arrow keys have no effect on Auto Transmit. Default is blank.

Status Line Visible

When Status Line Visible is checked, the status line is displayed on the last viewable row of the mobile client's screen. Default is blank.

Auto Enter

When Auto Enter is checked, all data will be sent to the host when all fields are completed. Valid for Block and Line mode. Default is blank.

Support TN3270E

When Support TN3270E is enabled, only Basic TN3270E [1] host requests are supported. The default value is "checked" or enabled. Please note that Full 3270E is not supported by RFTerm. Default is Enabled.

---

[1] As defined in Section 9 of RFC 2355.

Compatibility Tab

Figure 4-14 3270 Configuration – Compatibility Tab

- From the application menu, select **Session | Configure** or tap the **Configure** button on the ToolBar.

- A **Configure** dialog box will be displayed.

- Choose the **Connection** tab.

- Select 3270 as the Host Type, press the **3270 Config** button.

- Choose the **Compatibility** tab.

Parameters

Clear to EOL

The default value is Disabled (unchecked). Enable (check) this parameter to force the RFTerm 3270 TE to clear displayable text to the end of line or to the end of field attribute when writing displayable text from the host. Hidden attributes are removed.

Font

Figure 4-15 Configuration – Font Tab

- From the application menu, select **Session | Configure** or tap the **Configure** button on the ToolBar.

- A **Configure** dialog box will be displayed.

- Choose the **Font** tab.

Parameters

The default font is "Courier New" style "Normal". The default font may change based on the operating system and installed fonts on the mobile client.

Font

The drop down box contains the name of a fixed-width True Type font that is used to display the text on the screen. The fonts listed are pre-installed in the mobile client or installed by the end-user.

Style

This is the weight setting of the selected font. Available weights are: Default, Thin, Light, Normal, Medium, Bold, XBold, Heavy. The default is Normal.

Pointsize

Allows the pointsize of the selected font to be specified. The default is 10, valid values are from 6 to 72. If the **Specify Rows and Columns** checkbox is checked, Pointsize is calculated by RFTerm and this control is disabled (grayed out).

Specify Rows and Columns

The default is disabled (unchecked). Select this option to specify the rows and columns to fit the display using the Rows: and Cols: boxes. Using the specified rows and columns, RFTerm calculates the pointsize for the fonts. When this option is selected, the Pointsize control is disabled.

DoubleByte Fonts

RFTerm supports four Doublebyte Font sets:

- Traditional Chinese
- Simplified Chinese
- Japanese
- Korean

DoubleByte Fonts and Mobile Clients

Mobile devices handle the internal requirements of doublebyte fonts differently. The differences are based on the parameters of the operating system installed on the mobile clients. Some fonts are pre-installed with the Windows operating system, additional fonts are to be installed by the end user and frequently, additional font installation is a part of a packaged software installation.

RFTerm supports Doublebyte fonts on the following LXE mobile clients:

Client	Windows OS	Doublebyte Font Management
HX2, MX3X, MX5, MX7, VX3X, VX6, VX7	CE	One doublebyte font is available from LXE when ordering the device. LXE installs the font. Font is supported by RFTerm.
MX6	Windows Mobile	RFTerm installs and supports one of the Doublebyte fonts.
VX5	XP	End user installs a licensed font. Font is not available from LXE. Font can be accessed by RFTerm.

Colors

Traditional Windows color scheme is the default. Predefined color schemes are available to enhance visual readability of the display text on the mobile client. Further, mobile clients may be used in different lighting environments.
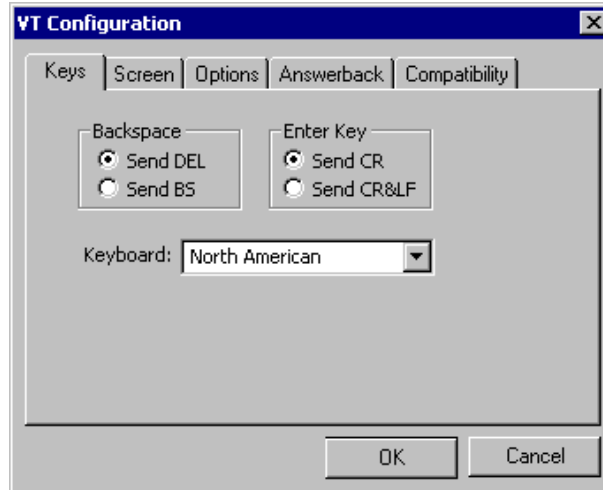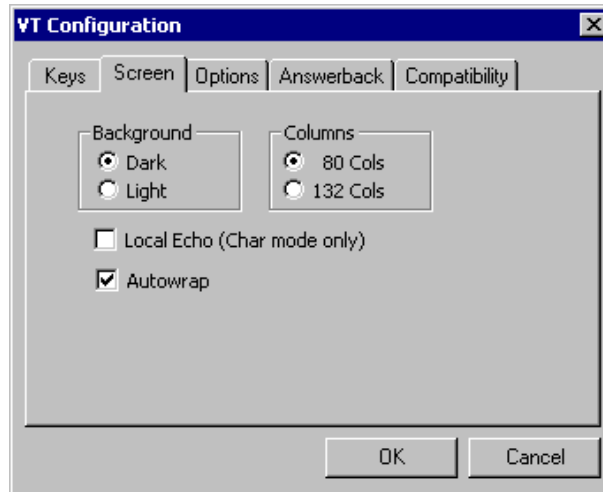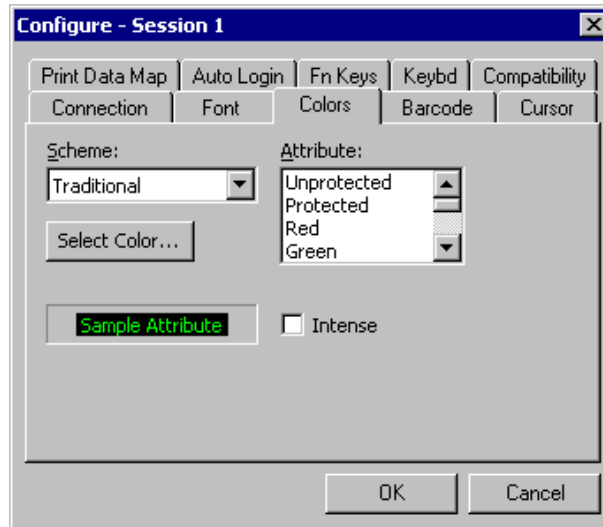

Figure 4-16 Configuration – Colors Tab

- From the application menu, Select **Session | Configure** or tap the **Configure** button on the ToolBar.

- A **Configure** dialog box will come up.

- Choose the **Colors** tab

In addition to predefined color schemes, a "Custom" color scheme may be used to configure various display attributes to suit the user's personal preferences.

Parameters

Scheme

To select a predefined color scheme, choose one from the Scheme drop down box. The sample box will change to reflect the current selected attribute foreground and background colors. The following pre-defined color schemes are available.

Traditional	This is the traditional IBM terminal "green screen" color scheme
Black on White	Used primarily on devices with monochrome LCD displays
White on Black	Inverse of black on white
Factory	This color scheme is optimized for factory lighting and color displays
Custom	This scheme allows the user to alter various attributes

Attribute

This is a list of mobile client display attributes, including Bold, for which colors may be changed as part of creating a Custom color scheme.

Sample Attribute

This is a sample box that shows the foreground and background color of the currently selected attribute.

Intense

This check box, when selected, applies the custom color to the Intense version of the selected attribute. This option applies to 3270 emulations.

Select Color

This button invokes the Color dialog to select a color for an attribute.

Figure 4-17 Color Dialog

How To: Custom Color Scheme

To create a custom color scheme, select the attribute from the attribute list box.

- Choose the attribute, whose color is to be changed.

- Tap the **Select Color.**.. button.

- The Color dialog appears.

- Select the desired color for the attribute. Click **OK**.

- The Sample box will change to reflect the newly selected color.

- For updating colors associated with the Intense mode of an attribute, check the Intense box then select the desired color.

15

Barcode

Barcode scanner options are provided for easy manipulation of data streams input through the serial port.

Note: Additional barcode manipulation features are available using the LXE Barcode Wedge for Windows CE based devices. Please refer to your device reference guide for additional details. If barcode manipulation is defined in both the Barcode Wedge and in RFTerm, the actions defined in the Barcode Wedge will be executed first, followed by actions defined in RFTerm. LXE recommends using the Barcode Wedge for barcode manipulation unless emulation-specific actions are required.
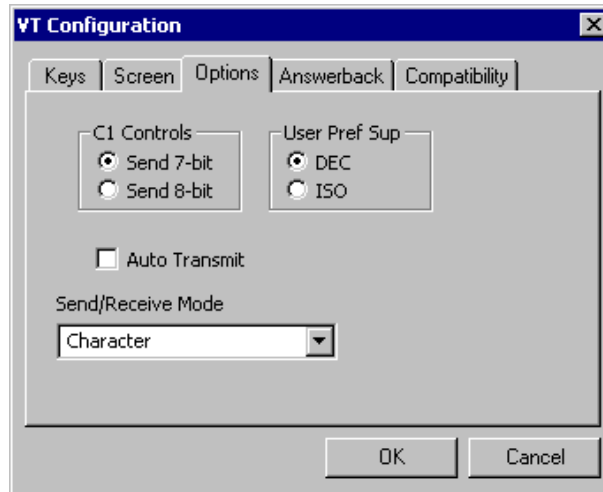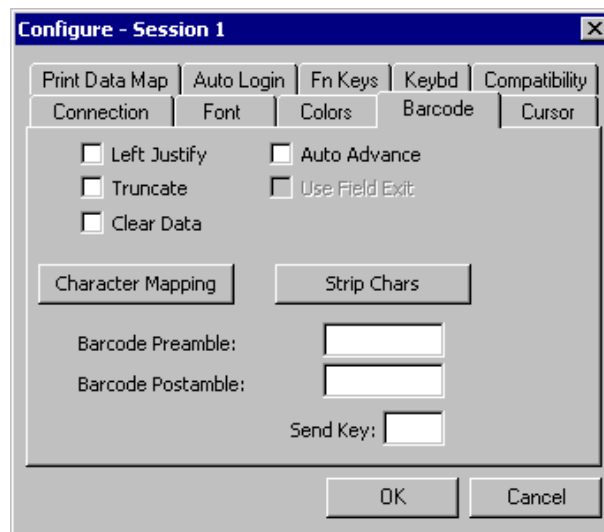

Figure 4-18 Configuration – Barcode Tab

- From the application menu, select **Session | Configure** or tap the **Configure** button on the ToolBar.
- A **Configure** dialog box will be displayed.
- Choose the **Barcode** tab.

Parameters

Left Justify

When checked, the Left Justify option shifts the cursor to the leftmost position of the current input field before any serial input data is displayed on the screen.

Truncate

The Truncate option instructs RFTerm to discard any serial input data that would be beyond the end of the current field.

Clear Data

The Clear Data option automatically erases all data from the current cursor location to the end of the current field before any serial input data is displayed.

Auto Advance

When checked, the Auto Advance option automatically moves the cursor to the beginning of the next input field after displaying all data input through the serial port.

Use Field Exit

The Use Field Exit feature is only available for *5250 emulations* and only when the Auto Advance box is checked. Checking both boxes will cause RFTerm to append Field Exit to the scanned barcode.

Barcode Preamble

This option will automatically insert the requested data at the head of the incoming data stream before it is displayed. Control characters and characters not accessible from the keyboard can be represented using the Hat Encoding table in Appendix B.

Barcode Postamble

This option will automatically insert the requested data at the tail of the incoming data stream before it is displayed. Control characters and characters not accessible from the keyboard can be represented using the Hat Encoding table in Appendix B.

Send Key

When checked, the Send Key option allows for a designated character in a barcode scan to automatically send an [Enter] command, provided it is the last printable character in the barcode.

Character Mapping

The Character Mapping option is used to setup a non-printable character to be incorporated into a barcode and interpreted by RFTerm as one of 16 predefined actions.

Character mappings are unique to each session, persistent when RFTerm is closed and reset if a client session setting is reset to its Default values. The default value for all programmable key codes is Accept.

For example, to map NULL to a Tab, click the down arrow in the Character text box. Select NULL. Click the down arrow in the Action text box. Select Tab. Click the Add button. The mapping list changes to show NULL mapped to Tab.
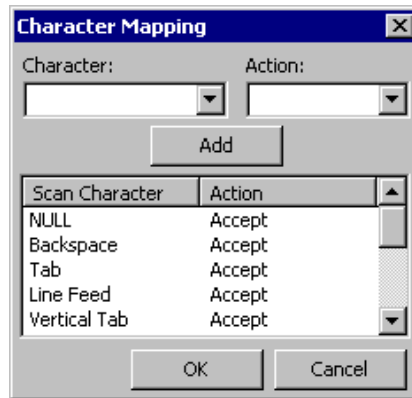
Figure 4-19 Barcode Character Mapping

Options available from the Action drop down menu are:

- Accept
- NULL
- Backspace
- Tab
- Line Feed
- Vertical Tab
- Form Feed
- Carriage Return
- Data Link Escape
- File Separator
- Group Separator
- Record Separator
- Unit Separator
- Enter (Send)
- Space
- Ignore (Drop)

Character Stripping

Figure 4-20 Barcode Character Stripping

Default values are Code ID = *, Pre-Strip = 0 and Post-Strip = 0. The asterisk (*) is a wildcard character. Click the OK button to save any changes/additions. Click the Cancel button or the X button to ignore any changes made.

Code ID Stripping

When Code ID Stripping is checked, the first character is stripped. When not checked, character stripping occurs based on the values in the Pre-Strip category.

Code ID

Code ID matches are based on this ID. The asterisk (*) is the default – which means match any character.

Pre-Strip

The Pre-Strip option removes the requested number of characters from the head of the incoming data stream before it is displayed.

Post-Strip

The Post-Strip option removes the requested number of characters from the tail of the incoming data stream before it is displayed.

Cursor

Cursor options are provided for easy identification of input fields. You may configure cursor type and auto-scrolling options for the session.
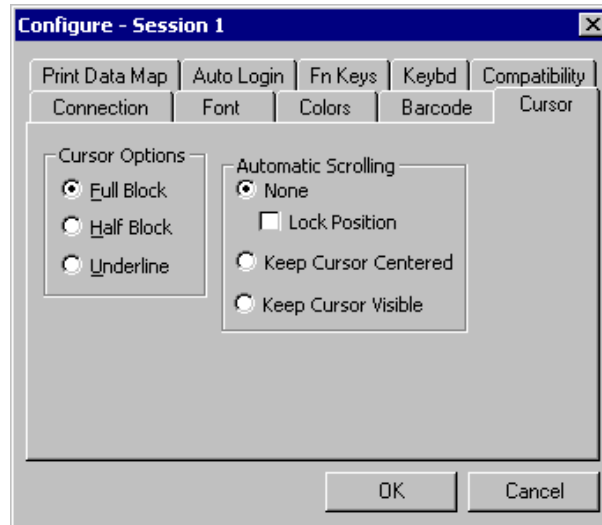
Figure 4-21 Settings – Cursor Tab

- From the application menu, select **Session | Configure**.
- A **Configure** dialog box will come up.
- Choose the **Cursor** tab

Parameters

Cursor Options

This option allows you to change the cursor appearance. The default is Full Block. Three options are available:

Full Block   The cursor appears as a full character block

Half Block   The cursor appears as a bottom half block

Underline   The cursor appears as an underscore line

Automatic Scrolling

This option enables automatic scrolling so the current cursor position is always visible. This option is helpful on mobile clients with small screens. The default is None. The following auto-scrolling options are available:

None	No automatic scrolling is allowed.
None – Lock Position	Lock the current scroll position in place. This checkbox is only active if Automatic Scrolling is set to None. The default value is blank. When it is checked, the user will not be able to move the screen from the current view. Upon restarting, the same "current" scroll position of the screen is shown.
Keep Cursor Centered	In this mode the cursor is always as close as possible to the center of the screen. When scrolling limits are reached, the cursor will move toward the edge of the client display.
Keep Cursor Visible	In this mode the cursor is always visible. The display is scrolled vertically and horizontally to prevent the cursor from moving out of view.

Print Data Map

Send and receive data on the same serial port on a per session basis, regardless of print setup. See also **File | Print Setup | Direct Serial**.
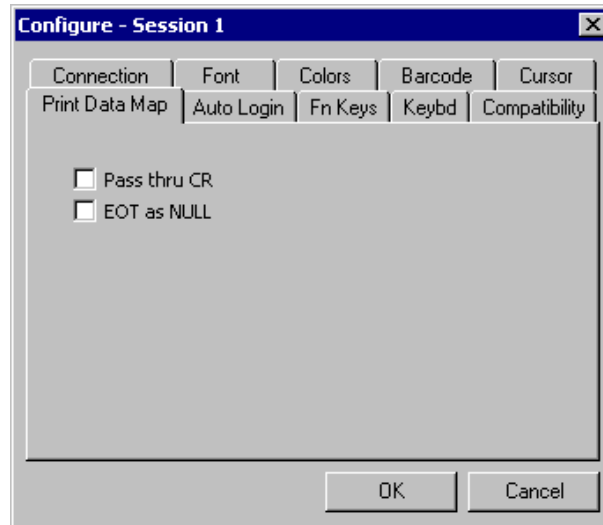
Figure 4-22 Configure - Print Data Map

- From the application menu, select **Session | Configure**.
- A **Configure** dialog box will come up.
- Choose the **Print Data** Map tab.

> ⚠️ The serial port configuration parameters are now located at **File | Print Setup | Direct Serial**.

Parameters

Pass thru CR

Enable (check) for pass through carriage return. Default is disabled (blank).

EOT as NULL

Enable (check) to replace EOT (End of Text) with a Null character.

The default is disabled (blank); EOT is a Space character.

Auto Login
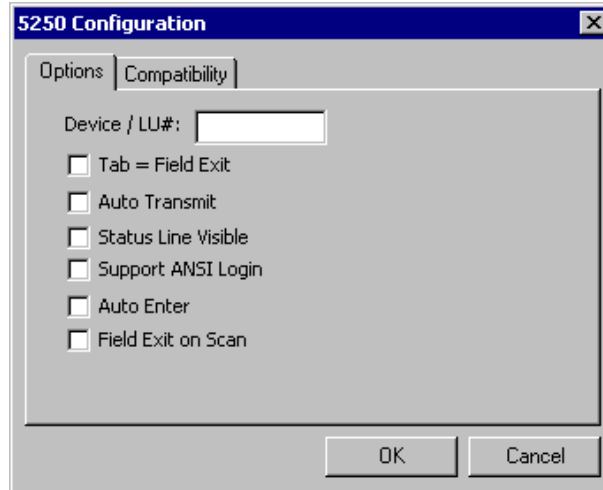
Use this option to create automatic login scripts.
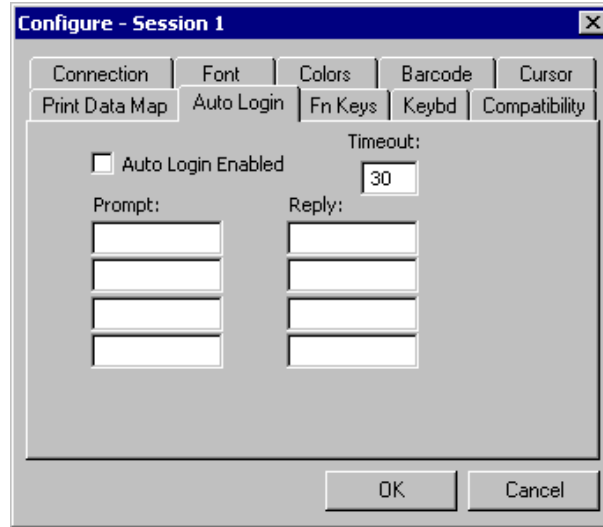
Figure 4-23 Configure – Auto Login Tab

- From the application menu, select **Session | Configure** or tap the **Configure** button on the ToolBar.
- A **Configure** dialog box will come up.
- Choose the **Auto Login** tab.

Parameters

Auto Login Enabled

When checked, RFTerm will automatically connect to this session the next time it is launched. If no prompt or reply is specified, RFTerm will simply wait at the client login screen, and proceed to connect to the next auto login enabled session. Should a session fail to connect, the client is prompted with a popup dialog requesting a retry or cancel. The default is Disabled.

Timeout

Enter how long RFTerm will wait for the session to connect and find the prompt string. The default is the standard RFTerm 30 second connection timeout.

Prompt

Enter the string to search for after RFTerm connects to the auto login enabled session. The prompts must be contained on successive screens or the reply will not be sent. The prompts are case sensitive. Leaving the Prompt blank but filling in data for the matching Reply will cause the Reply to be performed regardless of the screen data received from the host.

You can enter up to four Prompt/Reply pairs for 3270 and 5250 sessions and six Prompt/Reply pairs for ANSI sessions.

Reply

Enter the string, or action, to place at the current cursor location when RFTerm finds the Prompt string. Control characters and non-printable characters can be represented using the hat encoding table in Appendix B. The Replies are case sensitive. This data will only be sent to the host if the Reply ends in a carriage return. Maximum of 64 characters.

Note: For 3270 mobile devices, ATTN, CLR, ESC and SYSRQ must be contained within curly brackets { }.

Function Key Programming

Allows for programming all available function keys and the 4 arrow keys.
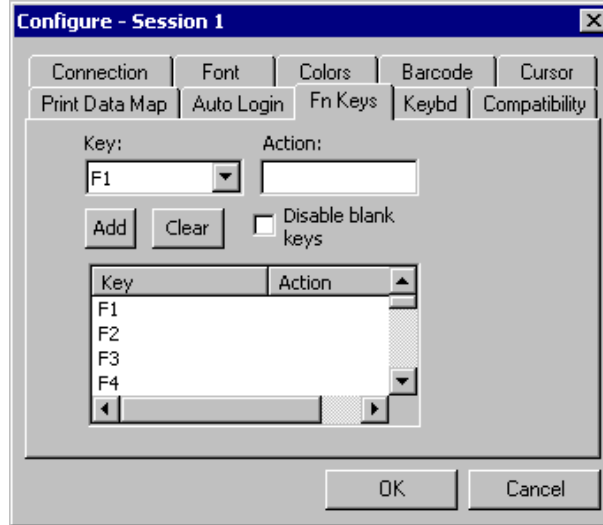
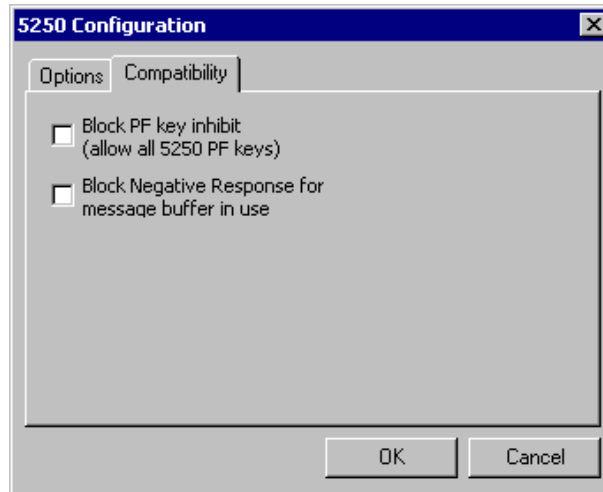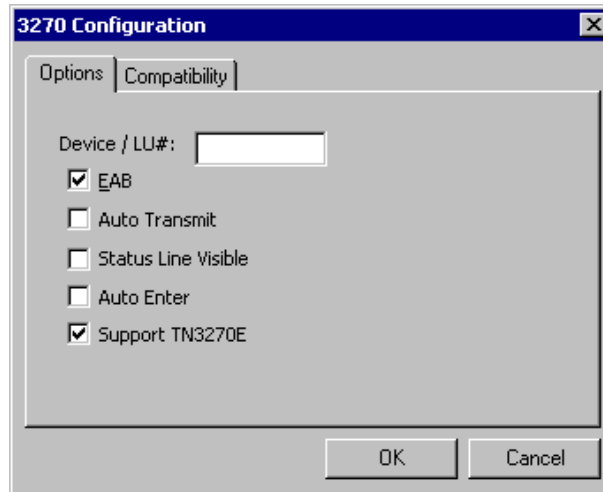**Figure 4-24 Configure - Function Keys**

- From the application menu, Select **Session | Configure** or tap the **Configure** button on the ToolBar.

- A **Configure** dialog box will come up.

- Choose the **Fn Keys** tab.

Parameters

Function Key Combo Box

Select the function key you wish to program. There are currently 40 selections for ANSI emulations and 24 each for 3270 and 5250 sessions. Most ANSI function keys have default values. The 4 arrow keys are also available for programming for all emulations.

Function Key Edit Box

Enter the keystrokes for that function key to perform. Control characters and characters not accessible from the keyboard can be represented using the hat encoding table in Appendix B. Maximum number of characters is 64.

Add

Adds the entry to the function key table.

Clear

Clears the entry in the function key table for the function key highlighted in the function key combo box.

Disable Blank Keys

The default value for Disable Blank Keys is Blank or empty. This option is available for ANSI, 3270 and 5250 emulations.

When the Disable Blank Keys checkbox is disabled (empty), all blank/unassigned function keys send nothing when pressed.

When the Disable Blank Keys checkbox is enabled (checked), then all blank function keys will send the default data for that function key.

Enable (check) the Disable Blank Keys checkbox to cause all blank function keys to send the default data for that function key.

Explanations:

- An unassigned key, such as F5, can send data assigned to another key, such as F18. Enable this value if you want unassigned keys to send nothing.

- In ANSI, an F5 keypress (when the F5 key is empty or unprogrammed) will send F18 data (the default value for an empty F5 key). The F18 data will be either the assigned default for an empty F18 or the user entered data for a non-empty F18.

Keybd
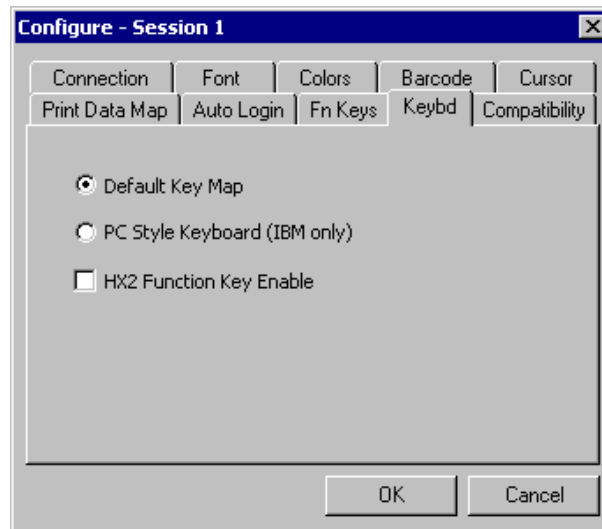
Figure 4-25 Configure - Keybd

- From the application menu, select **Session | Configure** or tap the **Configure** button on the ToolBar.
- A **Configure** dialog box will come up.
- Choose the **Keybd** tab.

Parameters

Default Key map

When selected, the current operating system default keyboard/keypad function key mapping is in effect.

PC Style Keyboard (IBM only)

When selected, function keys [Shift] [F1] through [Shift] [F12] will map to [F13] through [F24].

The default function key mapping is [Shift] [F1] through [Shift] [F4] will map to [F21] through [F24].

HX2 Function Key Enable

Note: Applies to HX2 with Alpha Mode 3 Tap keyboard only.

When unchecked, F11 – F19 [Blue] *plus* [1 – 9]) are used to implement RFTerm functions such as Connect/Disconnect, Hide Bars, etc. This behavior means a F11 – F19 keypress is not sent to the application, but instead performs a specified RFTerm function.

When checked, virtual keys F11 – F19 (Blue 1 *plus* 9) are no longer mapped to RFTerm functions but are mapped to F11 – F19. Other keypress combinations should be assigned by the user to the RFTerm function.

Compatibility

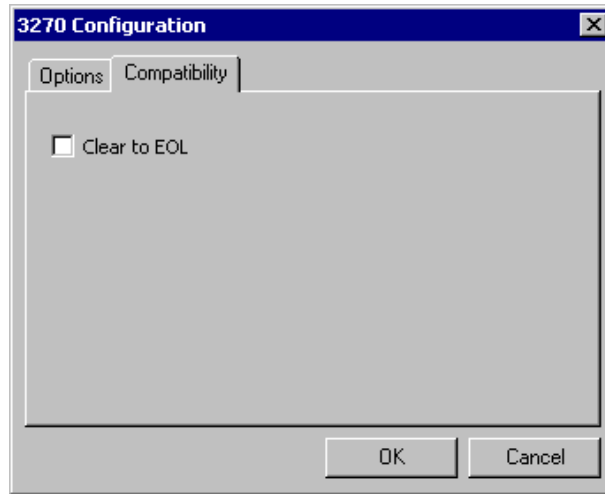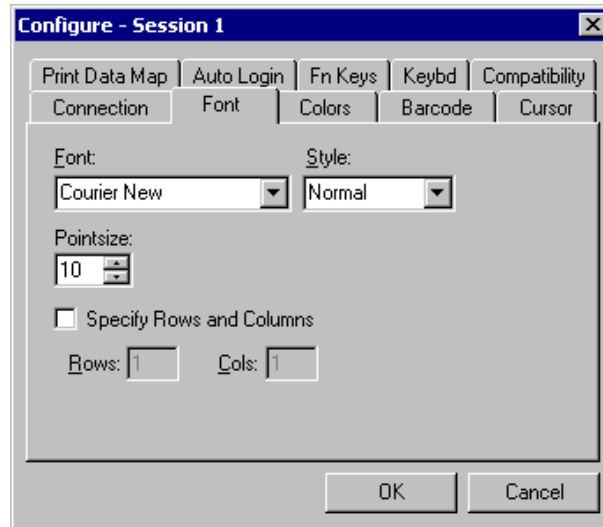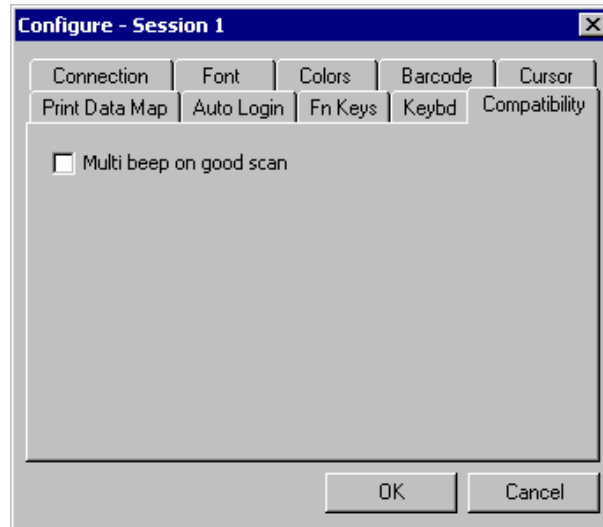**Figure 4-26 Configure – Compatibility**

- From the application menu, select **Session | Configure** or tap the **Configure** button on the ToolBar.

- A **Configure** dialog box will come up.

- Choose the **Compatibility** tab.

Parameters

Multi beep on good scan

Enables a multiple beep on a good scan. Default is disabled, a single beep ongood scan.

Chapter 5 Session Interaction

LXE RFTerm allows up to four simultaneous host client sessions. Although, each of these client sessions may be disconnected or connected, only one session is in the foreground at any time. The session in the foreground is the active session and receives all user interaction. Connected sessions in the background maintain their host session connections and update their (hidden) screen content.

Multiple Sessions

You can navigate between multiple sessions either from the application menu or by using the HotKey. The current active session is indicated in the application **Session** menu as shown below. Host addresses configured for each of the host sessions are indicated as part of the session names.
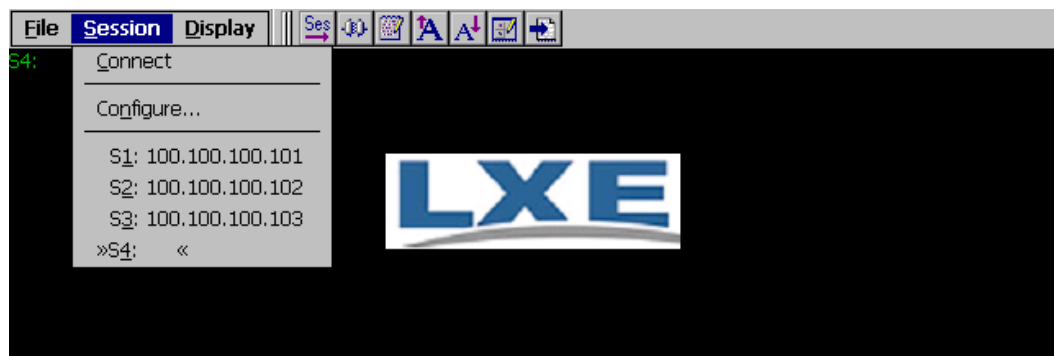
Figure 5-1 Session Menu

- Current Active Host Session is S4, and it is ready to be configured.

- Checkmarks or >> << indicate the current active host session. A checkmark indicates a connected session.

Tip: *To jump to the next connected host session; use the HotKey **[Ctrl] [Shift] [J]***

Password Protection

Access to the Session and Configure dialogs can be password protected.

- From the application menu, select **File | Settings**.
- A **Settings** dialog box will come up.
- Select the **Options** tab.
- Press the **Config Password** button.
- Enter the password and verify.

Figure 5-2 Change Password

Any attempt to open either the Settings or Configure dialogs will prompt the user for the password.
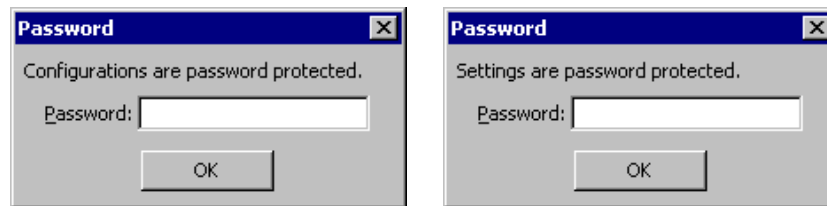
Figure 5-3 Configure Dialog /Configure Settings Password Dialogs

To remove password protection, first enter the password. Use the Backspace key to erase the password. Tap the OK button. If the password has been forgotten, please contact LXE Customer Support for assistance.

Connecting / Disconnecting from Host

To connect to a host, configure the client session parameters and select "Connect" from the application menu. Only disconnected sessions may be connected. Once connected, you may disconnect the host session by selecting a "Disconnect" from the application menu.

You must configure the host session prior to attempting a connection.

From the application menu select **Session | Connect** to connect or **Session | Disconnect** to disconnect. You may also tap the "Connect / Disconnect" button on the ToolBar.

Device Name In Use

When RFTerm receives an incoming host message that indicates the device name is in use, a "Device Name in Use" dialog box is displayed. The user clicks the OK button or the X button to close the dialog box.

SmartPads

Depending upon the terminal emulation type, certain keys may be required to navigate within the host application. The SmartPad is a soft keypad that provides access to all emulation specific keys. You can invoke the SmartPad only if the session is connected. The correct SmartPad appears for the emulation type of the current active session.

You can invoke the SmartPad from the application menu by selecting **Display | SmartPad**. You may also tap the "SmartPad" button on the ToolBar.

Tip: *You may use the SmartPad HotKey **[Ctrl][Shift][P]** to invoke this operation.*

SmartPads are not configurable dynamically.

Note: *This option is not available on Windows XP or Windows 2000 devices.*

3270 Host Key Descriptions

Attention

Attn - The Attention key interrupts the host application.

System Request

Srq - The System Request key gives context to the System Software such as the SSCP (System Services Control Point).

Clear

Clr - The Clear key causes a CLEAR Attention Identifier key to be sent to the host and the host responds according to the host application.

Reset

Rst - The Reset key resets the mobile client. Depending upon the current state, it removes the host application from a keyboard inhibit state, terminates the System Request functions and exits the mobile client from insert mode.

Erase Input

EIn - This key clears all unprotected input fields and moves the cursor to the beginning of the first input field.

Erase to End of Field

Eof - This key erases all data in an input field from the current cursor location to the end of the input field.

Delete

Del - The Delete key deletes data from an input field. When you press this key, the character at the cursor location is deleted, and all characters to the right of the cursor shift one position to the left.

Insert

Ins - This key toggles the insert mode on and off. In insert mode, characters are inserted at the current cursor location if space is available. The characters to the right of the cursor are shifted one character position to the right.

Home

Hm - This key repositions the cursor to the first input location of the first input field.

Tab or Next

Tb - This key moves the cursor forward to the next input field. When the cursor is not on an input field it moves to the next input field from the current cursor location.

Backtab or Previous

BTb - This key moves the cursor back to the previous input field. When the cursor is not on an input field, the cursor moves to the previous input field from the current cursor location. This key is equivalent to **[Shift][Tab]**.

Enter

Ent - The Enter key submits control to the host application.

New Line

NL - This function moves the cursor to the first input field on the next line.

Escape

Esc - The function is the same as the Reset command.

PA Keys

Pa1, Pa2, Pa3 - The PA1 through PA3 keys communicate with the host application. Their use is defined by the host application.

Function Keys

F1 – F24 - The Function keys F1 - F24 communicate with the host application. Their use is defined by the host application, unless their action has been programmed using the Function Key dialog.

5250 Host Key Descriptions

Attention

Attn - The Attention key interrupts the host application.

System Request

Srq - The System Request key gives context to the System Software such as the SSCP (System Services Control Point).

Clear

Clr - The Clear key causes a CLEAR Attention Identifier key to be sent to the host and the host responds according to the host application.

Reset

Rst - The Reset key resets the client. Depending upon the current state, it removes the host application from a keyboard inhibit state, terminates the System Request functions and exits the client from insert mode.

Erase Input

EIn - This key clears all unprotected input fields and moves the cursor to the beginning of the first input field.

Erase to End of Field

Eof - This key erases all data in an input field from the current cursor location to the end of the input field.

Field Minus

F- - This key causes the cursor to advance to the next field and a minus sign is inserted in the last position of a signed numeric-only field.

Field Plus

F+ - This key causes the cursor to exit an input field and insert null characters from the current cursor location to the end of the field.

Field Exit

FEx - This key behaves similar to the Field Plus key. It causes the cursor to exit an input field and insert null characters from the current cursor location to the end of the field.

Duplicate

Dup - This DUP key is used to insert DUP characters in a field for host processing.

Field Mark

Mrk - The Field Mark key is used to insert a Field Mark character in a field for host processing.

Delete

Del - The Delete key deletes data from an input field. When you press this key, the character at the cursor location is deleted, and all characters to the right of the cursor shift one position to the left.

Insert

Ins - This key toggles the insert mode on and off. In insert mode, characters are inserted at the current cursor location if space is available. The characters to the right of the cursor are shifted one character position to the right.

Home

Hm - This key moves the cursor to the first input location of the first input field.

Tab or Next

Tb - This key moves the cursor forward to the next input field. When the cursor is not on an input field it moves to the next input field from the current cursor location.

Backtab or Previous

BTb - This key moves the cursor back to the previous input field. When the cursor is not on an input field, the cursor moves to the previous input field from the current cursor location. This key is equivalent to **[Shift][Tab]**.

Enter

Ent - The Enter key submits control to the host application.

New Line

NL - This function moves the cursor to the first input field on the next line.

Roll Up

RUp - This function sends a roll up request to the host, similar to a scroll up.

Roll Down

RDn - This function sends a roll down request to the host, similar to a scroll down.

Cursor Select

Sel - This function allows the user to select an input field using the keyboard.

Print

Prt - This function prints the current screen.

Function Keys

F1 – F24 -The Function keys F1 - F24 communicate with the host application. Their use is defined by the host application, unless their action has been programmed using the Function Key dialog.

VT Host Key Descriptions

The following legend is used to indicate escape sequence values associated with various keys:

Bold Orange   Indicates 7 bit ASCII mode

Bold blue   Indicates 8 bit ASCII mode

Hold

Hld – This key has no current action.

Enter

Ent – The Enter or Return key transmits either a carriage return (CR) character or a carriage return and line feed (LF) character, depending on the VT configuration.

See Also: Chapter 6, figure titled "8 Bit Code Table."

Compose Character

Cmp – The Compose Character key does not transmit a code. Pressing the Compose character key starts a compose sequence which is used to generate characters that cannot be typed directly from the keyboard. This key is not implemented.

Tab

Tab – This key transmits a TAB character.

Delete

Del – This key transmits a DEL character.

Break

Brk – This key has no current action.

Cancel

Can – This key transmits a CAN character.

Answerback

Ans – This key sends an ENQ request to the host.

Find

Fnd – This key transmits the escape sequence **ESC [1 ~** or **CSI 1 ~**

Insert Here

Ins – This key transmits the escape sequence **ESC [2 ~** or **CSI 2 ~**

Remove

Rem – This key transmits the escape sequence **ESC [3 ~** or **CSI 3 ~**

Select

Sel – This key transmits the escape sequence **ESC [4 ~** or **CSI 4 ~**

Previous Screen

Prv – This key generates an escape sequence **ESC [5 ~** or **CSI 5 ~**

Next Screen

Nxt – This key transmits the escape sequence **ESC [6 ~** or **CSI 6 ~**

Escape

Esc – This key sends an ESC character to the host.

Backspace

BS – This key sends a BS character to the host.

Line Feed

LF – This key sends a LF character to the host

PF Keys

PF1 – PF4 – The numeric keypad keys PF1 through PF5 keys transmit the following escape sequences. The default action of Function Keys 1 through 4 is identical.

PF Key	ANSI Mode	VT 52 Mode
PF1	SS3 P or ESC OP	ESC P
PF2	SS3 Q or ESC OQ	ESC Q
PF3	SS3 R or ESC OR	ESQ R
PF4	SS3 S or ESC OS	ESQ S

The 5 keys F1-F5 on a VT client are local function keys and do not send codes. When a device has physical keys for F1-F5, we send PF1-PF4 for the corresponding F1-F4 and send a custom escape sequence for F5, the custom escape sequence depends on the device.

Function Keys

F1 – F40 – The Function keys F1 – F40 communicate with the host application. They transmit the following default escape sequences or their 7-bit equivalent, unless reprogrammed by the user in the Function Key dialog.

Function Key	Default Value
F1	ESC OP
F2	ESC OQ
F3	ESC OR
F4	ESC OS
F5	*none*
F6	CSI 17~
F7	CSI 18~
F8	CSI 19~
F9	CSI 20~
F10	CSI 21~
F11 (ESC)	CSI 23~ (ESC in VT100 and VT 52)
F12 (BS)	CSI 24~ (BS in VT 100 and VT 52)
F13 (LF)	CSI 25~ (LF in VT 100 and VT 52)
F14	CSI 26~
F15 (HELP)	CSI 28~
F16 (DO)	CSI 29~
F17	CSI 31~

Function Key	Default Value
F18	**CSI 32~**
F19	**CSI 33~**
F20	**CSI 34~**
F21	**CSI 25~**
F22	**CSI 26~**
F23	**CSI 28~**
F24	**CSI 29~**
F25	**CSI 31~**
F26	**CSI 32~**
F27	**CSI 33~**
F28	**CSI 34~**
F29 through F40	*none*

Configurable KeyBar

Most Pocket PC and CE devices have a limited keypad. They often rely upon their internal Soft Input Panel (SIP) to "spell type" data. However, the SIP usage is cumbersome and it does not provide any special host keys such as those needed in a terminal emulation program.

The configurable KeyBar allows users to customize a set of soft keys to invoke any host specific keys or emulator operations. Users can select from a set of pre-defined KeyBar templates. In addition, they can also configure up to four Custom KeyBars of their own. Users can navigate or "cycle" within a selected subset of KeyBars.

For CE devices, the KeyBar appears next to the application Menu and ToolBar. You can toggle it on or off by pressing the KeyBar icon next to the application menu. The emulator screen automatically adjusts to provide the maximum possible screen real estate.

Note: On the MX6, the KeyBar is shown above the menubar at the bottom of the screen. The ToolBar buttons are included with the KeyBar buttons. See Chapter 3 "RFTerm Settings", section titled "Display" for more KeyBar and ToolBar information when using an MX6.

To configure the KeyBar, please refer to the Options section under Configuration. Take some time to customize the KeyBar and become familiar with its use, it will be time well spent.

When enabled, the KeyBar appears on "top" of either the application menu, application ToolBar or SIP, depending upon their visibility state. Tap the arrow button on either end to cycle backward or forward through the configured KeyBars.

There are several pre-defined KeyBars that are provided with the application. These serve as emulation specific or operation specific templates. You may use these in addition to the customized KeyBars. The following table shows these pre-defined KeyBar templates.

See Also

The bitmaps and associated action text for all the actions and operations that can be associated with the configurable keys are located in "Appendix A – Key Maps", section titled "Icon / Button Identification".

KeyBar Name	KeyBar Buttons
Base Keys	← Ent Esc Tab Btb F₁ F₂ F₃ F₄ →
Special Keys	← OIA ▭ ▤ ◄ ► →
Scroll Keys	← ▣ ▣ ▣ ▣ ▣ ▨ ▤ ▶ →
F1 - F8	← F₁ F₂ F₃ F₄ F₅ F₆ F₇ F₈ →
F9 - F16	← F₉ F₁₀ F₁₁ F₁₂ F₁₃ F₁₄ F₁₅ F₁₆ →
F17 - F24	← F₁₇ F₁₈ F₁₉ F₂₀ F₂₁ F₂₂ F₂₃ F₂₄ →
F25 – F32	← F₂₅ F₂₆ F₂₇ F₂₈ F₂₉ F₃₀ F₃₁ F₃₂ →

KeyBar Name	KeyBar Buttons
F33 – F40	← F33 F34 F35 F36 F37 F38 F39 F40 →
IBM 1	← PA1 PA2 PA3 Er EOF Er INP Atn Rst Clr →
IBM 2	← Dup NL Hm SRq Ins Del Mrk Fld EXIT →
IBM 3	← Fld + Fld − Roll ↑ Roll ↓ Hlp Prt →
VT 1	← PF1 PF2 PF3 PF4 Fnd Ins Rem Sel →
VT 2	← Prv Nxt Hld Can Ans Del Esc Bsp →
VT 3	← LF Hlp Do U6 U7 U8 U9 U10 →
UDK11 - UDK18	← U11 U12 U13 U14 U15 U16 U17 U18 →
Digits 0 - 4	← 0 1 2 3 4 + − →
Digits 5 - 9	← 5 6 7 8 9 + − →

Figure 5-4 RFTerm KeyBars

Print Setup

You can configure a printer for RFTerm.

- From the application menu, select **File | Print Setup**.

This will bring up a popup dialog asking which print method you wish to use:
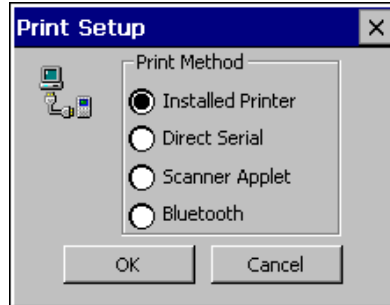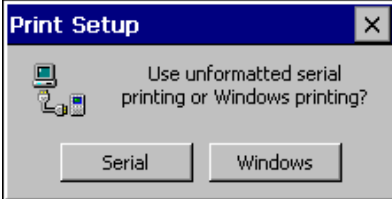
Figure 5-5 Print Setup

Note: If the mobile client supports Scanner Wedge output, the third option, Scanner Applet, can be selected. For mobile clients that do not support Wedge output, the Scanner Applet option is not available.

⚠️	Previous versions may display an alternate menu. The **Serial** button is similar in function to the **Direct Serial** button on the newer interface. The **Windows** button is similar in function to the **Installed Printer** button on the newer interface.

Installed Printer

If you select Installed Printer, a standard Microsoft Windows Print dialog box will appear.

Direct Serial

If you select Direct Serial, the following dialog is displayed:
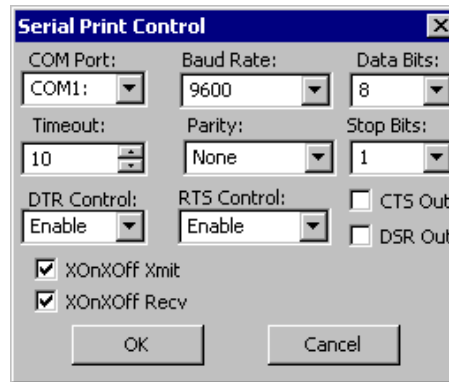
Figure 5-6 Serial Print Control

This dialog contains parameters that are identical, except for the Pass thru CR checkbox, to the Serial Data dialog; see Chapter 3 for Serial Data parameter details.

Select the external COM Port for the device and click OK. The device must have a serial printer attached to this COM port.

All RFTerm print commands will then send unformatted data to the serial printer. A screen print can be performed by using the keyboard shortcut [Ctrl] [Alt] [P].

> ⚠ The **Pass Thru CR** option is not located at **Session** | **Configure** on the **Print Data Map** tab.

Scanner Applet

If the mobile client running RFTerm supports Wedge Output, the third option, Scanner Applet, can be selected.

Not all COM port parameters supported in RFTerm can be supported by the mobile client Wedge due to the small form factor of the client.

Bluetooth

When the mobile device supports Bluetooth, the **Print Setup** tab contains a selection for a Bluetooth printer.

Com Port Error Dialog

Should RFTerm be unable to apply any of the COM Port settings, the following popup dialog will appear:
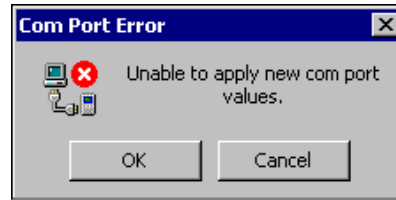
Figure 5-7 COM Port Error Dialog

Cradle Printing via IR

When a mobile client with an IR port is docked in a cradle and COM2 is selected as the serial print port on the mobile client, RFTerm immediately takes control of the port and continues to listen for print requests from the mobile client. RFTerm releases COM2 when the connection between the mobile client's COM2 port and the cradle is disconnected. While RFTerm is polling and controlling the cradle connection, the connection cannot be used by another application.

Media Copy Command

The serial port is opened on the initial MC command, all data received is written and the serial port is closed upon receipt of the proper MC command. Before the data is written to the serial port, the data is examined for carriage returns (CR / CR LF) within 80 bytes of the previous data stream. If found, the counter is reset and line wrapping/dropping does not occur.

Out of Range

Should the mobile client lose radio contact with the host, RFTerm will attempt to reconnect the active session. To enable this feature, check the Retry on Signal Loss box on the Connection tab of the Configure dialog.

If a connection cannot be established, the user is prompted with a "Host Not Responding" dialog box and given the option to Retry or Cancel the connection attempt.

Note: You will not have a Cancel button if the application is locked down by disabling application exit and/or by hiding the start menu. You must either re-establish the connection to the host or have the administrator intervene.

Text InputTool

Note: This option is not available on Windows XP or Windows 2000 devices.

The Text InputTool sends complete text strings to the display window at the current cursor location. A user can cache a set of alphanumeric text strings in the input tool.

You can invoke the InputTool from the application menu by selecting **Display | InputTool**. You may also tap the "InputTool" button on the ToolBar.
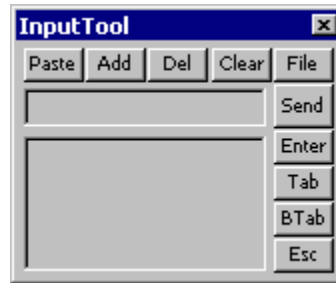
Figure 5-8 Text InputTool

Tip: You can also invoke the Text InputTool by using the HotKey [Ctrl][Shift][I].

You can build your cached list of text strings by pasting text from the clipboard (For example *[Edit][Copy]* in Microsoft WordPad) directly into the Text InputTool edit box. You can also directly type a text string into the edit box using the system soft keypad. Lastly, you can initialize this list from an ASCII text file.

Command	Description
Paste	Tapping the Paste button will copy any text from the clipboard into the edit box.
Add	Tapping the Add button adds any text in the edit box to the cache list. If the text string is already is in the list, it is not added again. Blank strings are not added.
Del	Tapping the Del button deletes the currently selected text string from the cache list.
Clear	Tapping the Clear button clears the edit box.
File	Tapping the File button opens a dialog to select a text file that can be read to initialize the list. Each text string to be added to the list must be on a separate line delimited by a CRLF. The following shows a sample input text file:

```
78438-8889-9494
Any text string here
Username
```

It is recommended that you create a file for initializing the list on the desktop and then ActiveSync it to the mobile client. Each session has its own list.

Send	The Send button sends the currently selected text string to the current cursor location on the client display.

Command	Description
Enter	Sends a carriage return keystroke to the TE application.
Tab	Sends a Tab keystroke to the TE application.
BTab	Sends a back tab keystroke to the TE application. This is equivalent to [Shift][Tab].
Esc	Sends an Escape keystroke to the TE application.

Each entry in the list is preceded by an index value in the form "1". This index, including the first space after the period, is stripped before sending. You can edit an existing list entry by selecting it, making your changes, then pressing Add. The saved list entry replaces the existing entry contents with the new value. If there is no leading index value, for example on a new entry, then Add will append a new list entry.

To create an entry, which begins with text similar to an index, first create a dummy entry, then edit it to include your desired text. For example, to create an entry with the contents "1. My text", first enter the text "new" and press Add. Assume that this becomes the third entry. Select the entry from the list and change "3. new" to "3. 1. My text", then press Add again.

The index values are used to identify the text sent with the "Input *n*" keys on the KeyBar. Be careful when deleting entries. Doing so may change the text sent via an Input key.

Special characters can also be entered in the text. This can be especially useful under VT emulation when custom escape sequences are required. The following special characters are available:

The Text InputTool also accepts hat-encoded characters, see Appendix B.

Operator	Meaning
\e	ESC character/action
\n	Newline character/action
\r	Enter character/action
\t	TAB character/action
\xDD	Hexadecimal value of byte

HotSpots

A HotSpot is a field on the client screen where a user can tap with a stylus to execute a function. This allows a user to interact with a host application without using physical keys or the KeyBar. A simple example might be the use of PF Keys. An operation associated with a PF key might be displayed on the client as "PF1 = Help". RFTerm emulators automatically detect this as a HotSpot and will send a PF1 key to the host when you tap on the PF1 text on the client display. To invoke a HotSpot, tap anywhere on the text of that HotSpot. HotSpots are not user configurable.

HotSpots are supported for 3270, 5250 and VT emulations. The emulations share some forms of HotSpots, such as function keys and menus, but others depend on the type of emulation.

The following table lists some text strings that are recognized as HotSpots.

HotSpot String	Sends
PFx=	Function key x. where x is between 1..9
PFxx=	For 3270, 5250 emulations: Function key x. where x is between 10..24
	For VT emulations: Function key x. where x is between 10..20
xx.	Menu option xx. where xx is any one or two digit number
Fxx=	Function key x. where x is between 1..24
<Fxx>=	Function key x. where x is between 1..24
Enter	Sends Enter key
"X. Menu choice"	Selects menu choice X
Double Tap	Sends Enter key
+	Roll Up key (5250 only)
-	Roll Down Key (5250 only)
More	Roll Down AID (5250 only)
Bottom	Roll Down AID (5250 only)

Note: Prior to using a "menu choice" HotSpot, the cursor must be in the input field where the menu choice is submitted. (Applies to IBM emulation only.)

Note: Reprogramming the functions keys using the Fn Keys tab under the Configure dialog alters the operation of a function key HotSpot.

Tip: HotSpots are sensitive to the stylus calibration on the device. To get accurate HotSpot taps, make sure your touchscreen is correctly calibrated.

Screen Panning

Most host applications have been designed for the desktop screen form factor. When viewing these applications from a handheld device, only a portion of the host screen is visible because of the smaller display of the handheld device. Horizontal and vertical scroll bars are typically used to set the virtual screen location. As an alternative to scroll bars, the screen can be positioned by "dragging" the stylus.

Imagine the host application screen as a large sheet under the display on the handheld device. Screen Panning allows users to "tap" and "drag" this sheet in any direction to move hidden areas of the host display into view on the handheld device without the use of scroll bars.

Follow these steps to use panning:

- Connect to a host application
- With the stylus, "tap and hold" anywhere on the client screen. Be careful not to tap on a HotSpot, if you do not also want a HotSpot action
- Drag the stylus, without lifting, to drag hidden areas of the screen into view

There is no configuration for Screen Panning. It is enabled whenever HotSpots are enabled for touch screen devices.

Access Control / Device Lockdown

In some cases, there is a need to prevent users from being able to change the emulator configuration. Further there may be a requirement to prevent users from exiting the TE application or launching other applications on the device. The Access Control features address these needs. Refer to the **File | Settings | Options** configuration section for details on configuring these features.

Note: RFTerm lockdown is available in Windows CE devices. Lockdown is not available in Windows 2000 devices. Windows XP devices can be locked down by configuring the OS.

Access control functionality allows for the following:

- Administrators can password protect access to the RFTerm session configuration settings (exceptions are listed in the above Note). This prevents changing the configuration:
  - From within the application
  - By loading another session
  - Saving the current session (to then load onto another device)
- Prevent users from invoking another application by disabling the Start menu. This prevents users from invoking another application while running the Terminal Emulation program.
- Prevent users from exiting the emulator by disabling the application exit feature.

HotKey Shortcuts

RFTerm has several predefined HotKeys. These may normally be used with devices that have a keypad. The following table lists the default HotKeys.

HotKey	Description
[Ctrl][Shift][N]	Configure dialog box
[Ctrl][Shift][E]	Settings dialog box
[Ctrl][Shift][C]	Connect to the host
[Ctrl][Shift][Down Arrow]	Decrease font size
[Ctrl][Shift][D]	Disconnect a *connected* host session
[Ctrl][Shift][H]	Display or Hide horizontal scroll bar
[Ctrl][Shift][P]**	Display or Hide SmartPad
[Ctrl][Shift][I]**	Display or Hide Text InputTool
[Ctrl][Shift][L]	Display or Hide all bars (Menu, Tool, Key)
[Ctrl][Shift][M]**	Display or Hide MenuBar
[Ctrl][Shift][T]	Display or Hide ToolBar
[Ctrl][Shift][K]	Display or Hide KeyBar
[Ctrl][Shift][V]	Display or Hide vertical scroll bar
[Ctrl][Shift][B]**	Display all bars at the bottom
[Ctrl][Shift][X]	Exit the emulator
[Ctrl][Shift][O] or [Alt][T]	Hide and unhide status bar (OIA)
[Ctrl][Shift][Up Arrow]	Increase font size
[Ctrl][Shift][S]	Invoke the session save dialog.
[Ctrl][Shift][J]	Jump to next *connected* host
[Shift][Down Arrow]	Scroll down one row
[Shift][Left Arrow]	Scroll left one column

HotKey	Description
[Shift][Right Arrow]	Scroll right one column
[Shift][Up Arrow]	Scroll up one row
[Ctrl][Down Arrow]	Scroll down one row segment (*see Note*)
[Ctrl][Left Arrow]	Scroll left one column segment (*see Note*)
[Ctrl][Right Arrow]	Scroll right one column segment (*see Note*)
[Ctrl][Up Arrow]	Scroll up one row segment (*see Note*)
[Ctrl][Shift][1]	Toggle between bold and normal font
[Ctrl][Alt][P]	Print the screen of the active session
[Ctrl][H]	Display frequently used shortcuts.

** Not supported on Windows XP/2000 devices.

Note: Depending on mobile client screen and font size constraints, the segment area used by a [Ctrl+Arrow] key sequence is approximately one-quarter of the total area available.

Automatic Session Startup

You can automate startup of up to four host sessions when the TE application starts. To do so, enable the "Auto Login Enabled" box in the Session | Configure | Auto Login tab, for each of the sessions that you want to automatically start. Each of these sessions will attempt a connection to the configured host when the emulator is started. If a connection cannot be established, the following popup dialog will appear, indicating the session number:
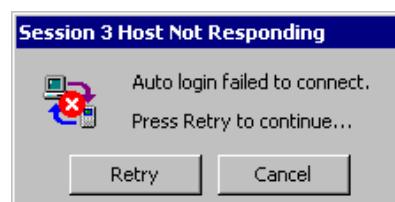
Figure 5-9 Auto Login Failure

Chapter 6 ANSI

ANSI Client Overview

Block Mode Communication

Block mode communication is a communication mode that enables the client to send an entire block of data to the host at one time. (Refer to section titled "Sending the Client Screen to the Host" found later in this chapter for more details.) Block mode greatly reduces RF traffic by sending data in this way. Block mode minimizes RF overhead better than the other modes. Thus, block mode supports more users on an RF channel and allows faster response time.

Local echo is a process that allows each character to display locally at the client as the client user presses the keys. The host does not see these characters until the user presses the [Enter] key.

Note: *In most cases, operations using programming tools that include screen painters cannot use block mode communication. Screen painters or designers usually come with programming tools such as Oracle, Ingress, Informix, and Powerhouse, which normally use character mode communications.*

Block Mode Setup

Refer to the following rules for setting up block mode communication:

- The programmer must include the following commands in the command string for setting up block mode:

```
ESC[12l      (turn local echo on in SRM)
ESC[=1l      (turn block mode on in LXELM)
```

Note: *If local echo is off in send-receive mode (SRM), the client operates in character mode while the scroll or block mode setting in LXELM is ignored.*

- The programmer must send formatted screens to the client from the host.

Scroll Mode Communication

Scroll mode communication is a communication mode where the client allows data entry only on one line. The client user can view the rest of the form, but the cursor returns to the line the host placed the cursor in whenever the user tries to enter data.

Scroll mode has the following advantages:

- Provides local protection of unformatted text.

- Protects all information on the screen except the command line, which is the line where the host positions the cursor. Character positions to the left of the cursor's current position are also protected.

- Enables the client user to move the cursor around the screen with the arrow keys. If the user moves the cursor, with the arrow keys, away from the command line and then tries to enter data, the cursor returns to the command line before accepting the data.

Note: Applications that edit individual characters do not perform well in scroll mode.

Scroll Mode Setup

Refer to the following rules for setting up scroll mode communications:

- The programmer must include the following commands in the command string for setting up scroll mode:

```
ESC[12l(turn on local echo in SRM)
ESC[=1h     (turn on scroll mode in LXESM)
```
- The programmer must construct a form consisting of one unprotected field on one line.

Initialization

The ANSI TE powers up in char mode and stays that way until the host changes the TE's mode. The host must specify that scroll or block mode is to be used during the current session.

Sending the Client Screen to the Host

Character Mode

Since each keypress is sent to the host as it is pressed, there is no concept of sending the screen in character mode. Pressing the [Enter] key in this case will simply send a carriage return and/or line feed to the host. No other sections of the screen are sent as a result of pressing [Enter].

Scroll Mode

Scroll mode allows user input on only one line of the display at a time until the user presses [Enter]. When this happens, the cursor will jump back to the active position if necessary. The cells between command start and command end will be sent to the host. The GATM/SATM/MATM modes are ignored in scroll mode. See the section titled "Scroll Mode Communication" found previously in this chapter for more details.

Block Mode

Block mode allows the user to input in any unprotected area on the display until the client user presses [Enter]. When this happens, the client will look at the current settings of GATM, SATM and MATM to determine exactly which screen cells are transmitted back to the host. The settings of these three modes yields 6 different combinations of cells (see Figure titled "Sending The Screen In Block Mode").

Protected cells are cells tagged with SPA (Start Protected Area) or DAQ 1 (Define Area Qualification).

Unprotected cells are cells that are neither SSA (Start Selected Area) nor SPA nor DAQ 1.

Note: DAQ 8 cells are considered unprotected.

			(x = don't care)		
Case #	Description of Cells Sent	Kind of Cells Sent	GATM	SATM	MATM
1	Entire screen	All	set 1h	set 17h	x
2	Current/next selected only	1 SSA	set 1h	reset 17l	reset 15l
3	All selected and all unprotected	SSA + !(SPA/DAQ1)	reset 1l	reset 17l	set 15h
4	Current/next selected and all unprotected	1 SSA + !(SPA/DAQ1/SSA)	reset 1l	reset 17l	reset 15l
5	All selected	SSA	set 1h	reset 17l	set 15h
6	All unprotected	!SSA + !(SPA/DAQ1)	reset 1l	set 17h	x

Figure 6-1 Sending The Screen In Block Mode

To use the table, first decide which of the 6 cases is desired. For example, to use case #6 set SATM and reset GATM. Since MATM is don't care in this case, it does not need to be set either way. Set up the screen so that it is marked protected or selected except where the client user will enter data. Now when the user presses [Enter], all cells that are not SSA, not SPA and not DAQ1 will be sent back to the host.

Sending the screen always operates on the virtual screen so EBM (Editing Boundary Mode) is ignored. Tab fields do not affect how the screen is sent.

When sending the screen, the "current/next" selected area means that if the cursor is currently on a selected area, that area will be sent. If not, search forward in the screen to find one and use that one. If none are found before the end of the screen, reset to the screen origin and look from there. If the client returns to the cursor position without finding one, it sends nothing.

In all cases the screen is scanned starting from the origin and each cell is transmitted in the order in which it is programmed on the display. For example, #5 means start at the origin and scan forward for all selected cells (type SSA only) sending them in the order they are found. For case #4, the current/next selected area is found first based on the current cursor position and it is remembered. However, the cells are still transmitted starting from the origin. The selected area is simply transmitted when the client comes across it in the display.

A single area separator (HT 0/9) byte is inserted into the transmit stream for each protected area on the screen (for case #3, 4, 5 and 6 only). However, if the first or last area on the screen is protected, no separator byte is inserted.

For example, assume case #3 is in effect. If the first few cells of the screen are defined as follows where p is a protected cell, s is an SSA selected cell and u is an unprotected cell:

    p1 p1 p1 s1 s1 s1 p2 p2 p2 s2 s2 s2 p3 p3 p3 s3 s3 s3 u1 u1 u1 ...

yields a transmit stream of:

    s1 s1 s1 HT s2 s2 s2 HT s3 s3 s3 u1 u1 u1...

Note: No separator is inserted for the first protected area on the screen. No separator appears between s3 and u1 since there are no protected cells between them.

If the last few cells on the same display are defined as follows:

    ...p4 p4 p4 p4 u2 u2 u2 u2 u2 p5 s4 s4 s4 s4 p6 p6

yields a transmit stream of:

    ...HT u2 u2 u2 u2 u2 HT s4 s4 s4 s4

Note: No separator is inserted for the last protected area on the screen.

To reduce traffic between the client and the host when sending the screen, use cases #3, #5 and #6. Tag most of the screen as protected by using SPA or DAQ1. Leave only the cells that the user will type/scan into as unprotected/selected. When the client user presses [Enter] to send the screen, only the unprotected/selected cells will be sent separated by tabs.

Tabbing

Three types of tabs are provided in the client:

- default
- area
- regular

Only one type of tab can be active at any time. Previously set tabs may be suppressed by a tab type with a higher priority. However, the suppressed tabs are still defined in the client.

Block mode must be active for the [Tab] key to work. If the [Tab] key is pressed in character mode, it will be transmitted back to the host. It is up to the host to echo it back to actually perform the tab. In scroll mode, pressing the [Tab] key is equivalent to pressing the [Spacebar] key.

The host may perform tabbing by sending the appropriate commands (i.e. HT) to the client regardless of the mode it is in.

When a tab command is received or the client user presses the [Tab] key, the cursor will move forward until the next tab stop is found. If none is found before the end of the display, the cursor will move to the origin of the virtual display and continue searching. The client will not tab to a tab that is in a protected area. Tabs are stationary and do not scroll when the screen does. EBM is ignored when the client tabs.

Default Tabs

These tabs are present when the client is first powered up. They appear on every line starting in column one and are spaced 8 columns apart (i.e. 1, 9, 17, etc.). These tabs have the lowest priority and they can be neither set nor erased by the host.

Area Tabs

These tabs can be set by the DAQ, SSA, EPA and ESA commands. They can be set by the host on any screen cell within the virtual display. These tabs have a medium priority. Setting one or more area tabs suppresses all of the default tabs. They can be erased only by the ED (Erase in Display) and EL (Erase in Line) commands when ERM (ERasure Mode) is set. If all of the area tabs are erased (and there are no regular tabs) the default tabs are restored.

Regular Tabs

These tabs can be set by the HTS, CTC and DAQ 7 commands. They can be set by the host on any screen cell within the virtual display. These tabs have high priority. Setting one or more regular tabs suppresses all default and area tabs. Regular tabs can be erased by the CTC, HTS and TBC commands. If all of the regular tabs are erased, any existing area tabs are restored. If no area tabs exist, the default tabs are restored.

ANSI Control Codes

Control codes enable a programmer to control or modify an LXE client and attached device. ANSI control codes consist of 7-bit and 8-bit control codes. This chapter provides information for the programmer on formatting control codes for LXE equipment. All the control codes supported by LXE are listed in this chapter. If a control code is not listed, LXE does not support it, and it is treated as a no operation (NOP).

Using the 8-Bit Code Table

The 8-bit code table has the same number of rows as the 7-bit code table, but twice as many columns. The codes on the left half of the table (columns 0 - 7) are used in a 7-bit and 8-bit environment. The eighth bit of these codes is 0. The codes on the right half of the table (columns 8 - 15) have an eighth bit of 1. These codes are used in an 8-bit environment and indirectly in a 7-bit environment.

The mnemonics in columns 0 and 1 are C0 (control 0) codes, which can be used in 7-bit environments. The mnemonics in columns 8 and 9 are C1 (control 1) codes, which can be used in 8-bit environments and indirectly in 7-bit environments as escape sequences.

A *code table* is a tool used to group all the characters in a character set with their codes.

A *mnemonic* is an abbreviated name that can be a single control character, escape sequence, or a control sequence.

8-Bit Code Table

Refer to the 8-bit code table below that lists all the decimal, hexadecimal, and octal values for each character:

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

◄— C0 Codes —► ◄———— GL Codes (ASCII Graphic) ————►

KEY

Character —— ESC [23 27 1B] —— Octal / Decimal / Hexadecimal
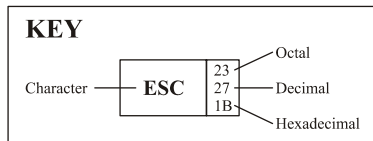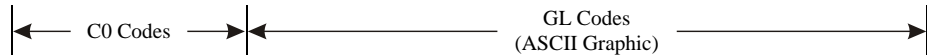
Figure 6-2 8-Bit Code Table, Columns 0 through 7

COLUMN →	8	9	10	11	12	13	14	15	BITS (b4 b3 b2 b1)	ROW
bits b8/b7/b6/b5	1 0 0 0	1 0 0 1	1 0 1 0	1 0 1 1	1 1 0 0	1 1 0 1	1 1 1 0	1 1 1 1		
0	200 128 80	DCS 220 144 90	//// 240 160 A0	° 260 176 B0	À 300 192 C0	320 208 D0	à 340 224 E0	360 240 F0	0 0 0 0	0
1	201 129 81	PU1 221 145 91	¡ 241 161 A1	± 261 177 B1	Á 301 193 C1	Ñ 321 209 D1	á 341 225 E1	ñ 361 241 F1	0 0 0 1	1
2	202 130 82	PU2 222 146 92	¢ 242 162 A2	² 262 178 B2	Â 302 194 C2	Ò 322 210 D2	â 342 226 E2	ò 362 242 F2	0 0 1 0	2
3	203 131 83	STS 223 147 93	£ 243 163 A3	³ 263 179 B3	Ã 303 195 C3	Ó 323 211 D3	ã 343 227 E3	ó 363 243 F3	0 0 1 1	3
4	IND 204 132 84	CCH 224 148 94	¤ 244 164 A4	264 180 B4	Ä 304 196 C4	Ô 324 212 D4	ä 344 228 E4	ô 364 244 F4	0 1 0 0	4
5	NEL 205 133 85	MW 225 149 95	¥ 245 165 A5	µ 265 181 B5	Å 305 197 C5	Õ 325 213 D5	å 345 229 E5	ö 365 245 F5	0 1 0 1	5
6	SSA 206 134 86	SPA 226 150 96	246 166 A6	¶ 266 182 B6	Æ 306 198 C6	Ö 326 214 D6	æ 346 230 E6	ö 366 246 F6	0 1 1 0	6
7	ESA 207 135 87	EPA 227 151 97	§ 247 167 A7	· 267 183 B7	Ç 307 199 C7	× 327 215 D7	ç 347 231 E7	÷ 367 247 F7	0 1 1 1	7
8	HTS 210 136 88	230 152 98	¨ 250 168 A8	270 184 B8	È 310 200 C8	Ø 330 216 D8	è 350 232 E8	ø 370 248 F8	1 0 0 0	8
9	HTJ 211 137 89	231 153 99	© 251 169 A9	¹ 271 185 B9	É 311 201 C9	Ù 331 217 D9	é 351 233 E9	ù 371 249 F9	1 0 0 1	9
10	VTS 212 138 8A	232 154 9A	ª 252 170 AA	º 272 186 BA	Ê 312 202 CA	Ú 332 218 DA	ê 352 234 EA	ú 372 250 FA	1 0 1 0	10
11	PLD 213 139 8B	CSI 233 155 9B	« 253 171 AB	» 273 187 BB	Ë 313 203 CB	Û 333 219 DB	ë 353 235 EB	û 373 251 FB	1 0 1 1	11
12	PLU 214 140 8C	ST 234 156 9C	254 172 AC	¼ 274 188 BC	Ì 314 204 CC	Ü 334 220 DC	ì 354 236 EC	ü 374 252 FC	1 1 0 0	12
13	R1 215 141 8D	OSC 235 157 9D	255 173 AD	½ 275 189 BD	Í 315 205 CD	Ÿ 335 221 DD	í 355 237 ED	ÿ 375 253 FD	1 1 0 1	13
14	SS2 216 142 8E	PM 236 158 9E	256 174 AE	276 190 BE	Î 316 206 CE	336 222 DE	î 356 238 EE	376 254 FE	1 1 1 0	14
15	SS3 217 143 8F	APC 237 159 9F	257 175 AF	¿ 277 191 BF	Ï 317 207 CF	ß 337 223 DF	ï 357 239 EF	//// 377 255 FF	1 1 1 1	15

←— C1 Codes —→ ←————— GR Codes (DEC Supplemental Codes) —————→

KEY

Character — ESC — 23 (Octal) / 27 (Decimal) / 1B (Hexadecimal)
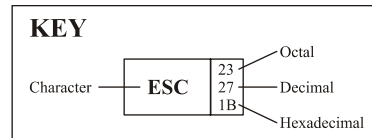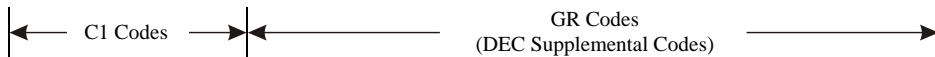
Figure 6-3 8-Bit Code Table, Columns 8 through 15

C0 Codes: 7-Bit Control Codes

C0 codes are 7-bit control codes that enable the programmer to perform basic functions with LXE clients and attached devices. This section lists all the C0 codes supported by LXE equipment. The examples in this chapter show applications for each control code. Not all C0 codes are supported by LXE equipment. If a code is not listed, LXE does not support it, and it is treated as a no operation (NOP).

BEL – Bell

Function

BEL causes the client to beep.

Hexadecimal Value	**07**
Control Code	**^G**
Modes that affect this code	**None**

Example

Refer to the following example on how a programmer can format a BEL command in the C programming language:

```
printf("Alert the user to this message!");
printf("\007\007\007\007");          /*Beep 4 times*/
printf("\007\007\007\007\007\007");    /*Beep 1 long beep*/
```

BS – Backspace

Function

BS moves the cursor one space to the left. The cursor stops at the left margin of the virtual display.

Hexadecimal Value	**08**
Control Code	**^H**
Modes that affect this code	**None**

Example

Refer to the following example on how a programmer can format a BS command in the C programming language:

```
printf("The cursor will backspace one space.");
printf("\010");                /*(Octal for hex 08)*/
```

CR – Carriage Return

Function

CR returns the cursor to the left margin (of the virtual screen) of the row that the cursor is currently in.

Hexadecimal Value	**0D**
Control Code	**^M**
Modes that affect this code	**None**

Example

Refer to the following example on how a programmer can format a CR command in the C programming language:

```
printf("Return cursor to beginning of \n");
printf("this second line.");
printf("\015");              /*Octal for hex 0D*/
```

DC2 – Device Control 2

Function

DC2 is ignored by the client device while media copy is enabled. The DC2 code will not be sent to the printer.

Hexadecimal Value	**12**
Control Code	**^R**
Modes that affect this code	**None**

Example

Refer to the following example on how a programmer can format an DC2 command in the C programming language:

```
printf("\033[5i");   /*Enables following text to copy to the
                       RS-232 port*/
printf("This message will copy to the RS-232 port.");
printf("\022");      /*Terminal ignores this code and does
                       not send it to the RS-232 port.*/
printf("\033[4i");   /*Disables RS-232 port*/
```

DC4 – Device Control 4

Function

DC4 is an alternative way to turn off media copy.

Hexadecimal Value	**14**
Control Code	**^T**
Modes that affect this code	**None**

Example

Refer to the following example on how a programmer can format an DC4 command in the C programming language:

```
printf("\033[5i");   /*Enables following text to copy to the
                        RS-232 port*/
printf("This message will copy to the RS-232 port.");
printf("\020");      /*Disables RS-232 port*/
```

DEL – Delete

Function

DEL erases a character in the data stream. DEL is for client user use only. The host does not send this command.

Hexadecimal Value	**7F**
Control Code	**[Delete]** *or* **[Ctrl] = [8]**
Modes that affect this code	**None**

Example

If an user enters a keystroke by mistake, the user can press [Del] or [Ctrl]+[8] causing the client to backspace, which deletes the last character.

ENQ – Inquiry

Function

ENQ requests a response from the client. The client responds with an answerback message.

Hexadecimal Value	**05**
Control Code	**^E**
Modes that affect this code	**None**

Example

Refer to the following example on how a programmer can format an ENQ command in the C programming language:

```
    printf("Requesting client response now.");
printf("\005");                /*Octal for hex 05*/
gets(input_str);               /*input_str =
                               LXE/q/08/TERMS67001C/0010*/
```

ESC – Escape

Function

ESC extends the set of ANSI controls by acting as a code extension.

Hexadecimal Value	**1B**
Control Code	**^[**
Modes that affect this code	**None**

Example

Refer to the following example on how a programmer can format an ESC command in the C programming language:

```
printf("\033E");               /*Octal for hex 1B. ESC E is a NEL
                               command*/
```

FF – Form Feed

Function

FF is processed as a line feed (LF) in LXE clients and advances the cursor down one line. Refer to the Line Feed control code for a more detailed description.

Hexadecimal Value	**0C**
Control Code	**^L**
Modes that affect this code	**Scroll Region, LNM**

Example

Refer to the following example on how a programmer can format an FF command in the C programming language:

```
printf("This appears on line 1.\n");
printf("\014");              /*(Octal for hex 0C)*/
printf("\014");
printf("This appears on line 4.");
```

See Also

LF

HT – Horizontal Tab

Function

HT moves the cursor to the next predetermined position (tab stop) to the right, within the virtual screen.

If there are no more tab stops after the cursor, it will jump to the origin of the screen and continue looking for the next tab stop from there.

Note: The cursor will not tab to any protected screen cells.

Hexadecimal Value	**09**
Control Code	**^I**
Modes that affect this code	**None**

Example

Refer to the following example on how a programmer can format the HT command in the C programming language:

```
printf("\033[1;25H");        /*CUP-place cursor at 1,25*/
printf("\033H");             /*HTS-Set Tab*/
printf("\033[1;1H");         /*CUP-place cursor at 1,1*/
printf("\011");              /*HT-tab to 1,25. Octal for hex
                             09*/
```

LF – Line Feed

Function

LF moves the cursor down one line.

If the cursor is outside the scrolling region when the Line Feed is received, it will not move past the bottom of the virtual screen. Scrolling does not occur outside the scrolling region.

If the cursor is on the bottom row of the scrolling region when the Line Feed is received and the client is in scroll mode, a blank line is inserted at the cursor and the screen scrolls up one (1) row.

Hexadecimal Value	**0A**
Control Code	**^J**
Modes that affect this code	**Scroll Region, LNM**

Example

Refer to the following example on how a programmer can format an LF command in the C programming language:

```
print("\033[20h");            /*Set LNM*/
printf("This appears on line 1.\n");
printf("\012");               /*Octal for hex 0A*/
printf("\012");
printf("This appears on line 4.");
```

NUL – Null

Function

NUL acts as a space-fill or time-fill character.

Note: You can insert a NUL anywhere in the data stream without affecting the information content.

Hexadecimal Value	**00**
Control Code	**^@**
Modes that affect this code	**None**

Example

Refer to the following example on how a programmer can format a NUL command in the C programming language:

```
printf("There will \000not be any\n");
printf("extra \000characters here.");
```

VT – Vertical Tab

Function

VT is processed as a line feed (LF) in LXE clients and advances the cursor down one line. Refer to the Line Feed control code for a more detailed description.

Hexadecimal Value	**07**
Control Code	**^G**
Modes that affect this code	**Scroll Region, LNM**

Example

Refer to the following example on how a programmer can format a VT command in the C programming language:

```
printf("This appears on line 1.\n");
printf("\013");                /*(Octal for hex 0B)*/
printf("\013");
printf("This appears on line 4.");
```

See Also

LF

C1 Codes: 8-Bit Control Codes

C1 codes are 8-bit control codes that enable the programmer to perform various functions with LXE clients and attached devices. Some environments operate in 7-bit code; therefore, 8-bit control codes also have 7-bit equivalents. This section lists all the C1 codes supported by LXE equipment. The examples show applications for each control code. LXE supports most of the C1 codes but not all of them. If a code is not listed, LXE does not support it, and it is treated as a no operation (NOP).

CSI – Control Sequence Introducer

Function

CSI indicates the start of a control sequence.

Hexadecimal Value	**9B**
7-Bit Equivalent	**ESC[**
Modes that affect this code	**None**

Example

Refer to the following example on how a programmer can format a CSI command in the C programming language:

```
printf("\033[10C");          /*CUF moves cursor 10 spaces
                             forward. 033[ represents hex 9B*/
```

EPA – End of Protected Area

Function

EPA indicates that the cursor's position is the end of a string of character positions protected from alteration by the client and guarded against transmission back to the host. The area from the cursor position forward is marked to accept all input as if a DAQ 0 command was issued from that position.

Hexadecimal Value	**97**
7-Bit Equivalent	**ESC W**
Modes that affect this code	**None**

Example

Refer to the following example on how a programmer can format an EPA command in the C programming language:

```
printf("\033V");              /*SPA*/
printf("This is the start of a protected area\n");
printf("This is the end of a protected area");
printf("\033W");              /*EPA*/
```

See Also

SPA (which precedes an EPA command)

ESA – End of Selected Area

Function

ESA indicates that the cursor position is the end of a string of character positions selected for transmission in a data stream to the host. The area from the cursor position forward is marked to accept all input as if a DAQ 0 command was issued from that position.

Note: The ESA does not initiate the RF transmission.

Hexadecimal Value	**87**
7-Bit Equivalent	**ESC G**
Modes that affect this code	**None**

Example

Refer to the following example on how a programmer can format an ESA command in the C programming language:

```
printf("\033F");            /*SSA-start of selected area*/
printf("This area is selected for transmission");
printf("\033G");            /*ESA-end of selected area*/
```

See Also

SSA (which precedes an ESA command).

HTJ – Horizontal Tab with Justify

Function

HTJ shifts the characters between the preceding tab position and the left of the cursor position, up to the next tab position. The screen cells to the left of the justified text will be space filled.

The cursor will also move to the left of the next tab stop. If the cursor was on a tab stop when the HTJ command was received, the cursor will move to the next tab stop with no justification. This command will act on regular tabs if they exist, otherwise it will act on default tabs. HTJ will not act on area tabs.

Hexadecimal Value	**89**
7-Bit Equivalent	**ESC I**
Modes that affect this code	**None**

Example

Refer to the following example on how a programmer can format an HTJ command in the C programming language:

```
printf("\033[5;5H");        /*CUP-move cursor to 5;5*/
printf("\033H");            /*HTS-tab set*/
printf("\033[5;12H");       /*CUP-move cursor to 5;12*/
printf("\033H");            /*HTS-tab set*/
printf("\033[5;5H");        /*CUP-move cursor to 5;5*/
sleep(10);                  /*Wait for user to enter data*/
printf("\033I");            /*HTJ-justify data to the  tab
                            set*/
gets(s);                    /*Wait for user to press <Enter>
                            and read in data*/
```

HTS – Horizontal Tabulation Set

Function

HTS sets a tab position at the cursor position.

Hexadecimal Value	**88**
7-Bit Equivalent	**ESC H**
Modes that affect this code	**TSM**

Example

Refer to the following example on how a programmer can format an HTS command in the C programming language:

```
printf("\033[4,20H");       /*CUP-move cursor to 4,20*/
printf("\033H");            /*HTS-Set Tab Stop*/
```

IND – Index

Function

IND moves the cursor down one row without changing column position.

If the cursor is on the bottom row of the scrolling region, the display within the region will scroll up by one (1) row.

No scrolling occurs if the cursor is outside of the scrolling region.

Hexadecimal Value	**84**
7-Bit Equivalent	**ESC D**
Modes that affect this code	**Scroll Region**

Example

Refer to the following example on how a programmer can format an IND command in the C programming language:

```
printf("\033D");              /*Index-move down 1 row*/
```

NEL – Next Line

Function

NEL moves the cursor to the left margin on the next line.

If the cursor is on the bottom row of the scrolling region, the display within the region will scroll up by one (1) row.

No scrolling occurs if the cursor is outside of the scrolling region.

Hexadecimal Value	**85**
7-Bit Equivalent	**ESC E**
Modes that affect this code	**Scroll Region**

Example

Refer to the following example on how a programmer can format an NEL command in the C programming language:

```
printf("\033E");              /*NEL-down one line to the left
                              margin*/
```

RI – Reverse Index

Function

RI moves the cursor up one line without changing the column position.

If the cursor is on the top row of the scrolling region, the display within the region will scroll down by one (1) row.

No scrolling occurs if the cursor is outside the scrolling region.

Hexadecimal Value	**8D**
7-Bit Equivalent	**ESC M**
Modes that affect this code	**Scroll Region**

Example

Refer to the following example on how a programmer can format an RI command in the C programming language:

```
printf("This prints on line 3.");
printf("\015");              /*Carriage Return. Move to column
                              1*/
printf("\033M");             /*RI-move up 1 line*/
printf("This prints on line 2.");
```

SPA – Start of Protected Area

Function

SPA indicates the start of a string of character positions protected from alteration by the client and guarded against transmission back to the host. This command is equivalent to DAQ 1.

Hexadecimal Value	**96**
7-Bit Equivalent	**ESC V**
Modes that affect this code	**None**

Example

Refer to the following example on how a programmer can format an SPA command in the C programming language:

```
printf("\033V");              /*SPA*/
printf("This is the start of a protected area\n");
printf("This is the end of a protected area");
printf("\033W");              /*EPA*/
```

See Also

EPA (which follows the SPA command)

SSA – Start of Selected Area

Function

SSA indicates the start of a string of character positions selected for transmission back to the host. This command is similar to DAQ 8, except that SSA allows user input whereas DAQ 8 does not.

Note: The SSA does not initiate the RF transmission.

Hexadecimal Value	**86**
7-Bit Equivalent	**ESC F**
Modes that affect this code	**None**

Example

Refer to the following example on how a programmer can format an SSA command in the C programming language:

```
printf("\033F");              /*SSA-start of selected area*/
printf("This area is selected for transmission");
printf("\033G");              /*ESA-end of selected area*/
```

See Also

ESA (which follows the SSA command)

ANSI Escape Sequences

This chapter provides information on coding ANSI escape sequences, which extend the functions available with the control codes. If an ANSI escape sequence is not listed, LXE does not support it, and it is treated as a no operation (NOP).

CBT – Cursor Backward Tabulation

Function

CBT moves the cursor left, to the preceding tab position within the virtual screen.

If there are no tab stops before the cursor, it will jump to the lower right corner of the screen and continue looking backward for the next tab stop.

Note: The cursor will not tab to any protected screen cells.

Escape Sequence	**ESC[PnZ**
Modes that affect this code	**None**

where:

Pn is the number of tab positions the cursor moves to the left.

Z is the final character.

Parameter Values

Refer to the following table for the movement of the cursor according to the value that you enter:

When you enter...	Then the cursor moves...
0	back one tab stop because the value defaults to 1 when you enter 0.
1 (default)	back one tab stop.
a number greater than 1	back the number of tab stops the value indicates within the bounds of the virtual screen.

Example

Refer to the following example on how a programmer can format a CBT command in the C programming language:

```
printf("\033[1;25H");        /*CUP-place cursor at row 1, column
                             25*/
printf("\033H");             /*HTS-set tab set*/
printf("\033[1;35H");        /*CUP-place cursor at row 1, column
                             35*/
printf("\033[1Z");           /*CBT-move cursor back to tab set*/
```

CHA – Cursor Horizontal Absolute

Function

CHA moves the cursor to the column (on the current row) that the parameter indicates. If a number greater that 80 is entered, the cursor will move to column 80 by default.

Note: Column number depends on the setting in Session Configuration.

Escape Sequence	**ESC[PnG**
Modes that affect this code	**None**

where:

Pn is the column number the cursor moves to.

G is the final character.

Parameter Values

Refer to the following table for the effect the value that you enter has on the cursor's position:

When you enter...	Then the cursor moves...
0	to column one because the value defaults to 1 when you enter 0.
1 (default)	to column one.
a number greater than 1	to the column the value indicates in the current line.

Example

Refer to the following example on how a programmer can format a CHA command in the C programming language:

```
printf("\033[7G");          /*move cursor to column 7 of the
                            current line*/
```

CHT – Cursor Horizontal Tab

Function

CHT moves the cursor right, to the next tab position within the virtual screen. If there are no tab stops found after the cursor, it will jump to the origin of the screen and continue looking for the next tab stop.

Note: The cursor will not tab to any protected screen cells.

Escape Sequence	**ESC[PnI**
Modes that affect this code	**None**

where:

Pn is the number of tab positions the cursor moves to the right.

I is the final character.

Parameter Values

Refer to the following table for the effect the value that you enter has on the cursor's position:

When you enter...	Then the cursor moves...
0	forward one tab position because the value defaults to 1 when you enter 0.
1 (default)	forward one tab position.
a number greater than 1	forward the number of tab positions that the value indicates within the bounds of the virtual screen

Example

Refer to the following example on how a programmer can format a CHT command in the C programming language:

```
printf("\033[1;25H");          /*CUP-place the cursor at row 1,
                               column 25*/
printf("\033H");               /*HTS-tab stop*/
printf("\033[1;1H");           /*CUP-move cursor up to  row 1,
                               column 1*/
printf("\033[1I");             /*CHT-place cursor horizontal tab*/
```

CNL – Cursor Next Line

Function

CNL moves the cursor down one or more lines to the left margin from the cursor's current position. The cursor will stick at the bottom margin if a large number is entered.

Escape Sequence	**ESC[PnE**
Modes that affect this code	**None**

where:

Pn is the number of lines the cursor moves down.

E is the final character.

Parameter Values

Refer to the following table for the effect the value that you enter has on the cursor's position:

When you enter...	Then the cursor moves...
0	to the left margin on the next line from its current position because the value defaults to 1 when you enter 0.
1 (default)	to the left margin on the next line from its current position.
a number greater than 1	down, as many lines as the value indicates, to the left margin.

Example

Refer to the following example on how a programmer can format a CNL command in the C programming language:

```
printf("\033[1;20H");        /*CUP-place cursor at row 1, column
                             20*/
printf("\033[1E");           /*CNL-move cursor to next line,
                             column 1*/
```

CPL – Cursor Preceding Line

Function

CPL moves the cursor up one or more lines to the left margin from the cursor's current position. The cursor will stick at the top margin if a large number is entered.

Escape Sequence	**ESC[PnF**
Modes that affect this code	**None**

where:

Pn is the number of lines the cursor moves up.

F is the final character.

Parameter Values

Refer to the following table for the effect the value that you enter has on the cursor's position:

When you enter...	Then the cursor moves...
0	up one line to the left margin because the value defaults to 1 when you enter 0.
1 (default)	up one line to the left margin.
a number greater than 1	up, as many lines as the value indicates, to the left margin.

Example

Refer to the following example on how a programmer can format a CPL command in the C programming language:

```
printf("\033[3;20H");        /*CUP-move the cursor to row 3,
                             column 20*/
printf("\033[2F");           /*CPL-move the cursor up 2 lines*/
```

CPR – Cursor Position Report

Function

CPR reports the cursor's position to the host. DECOM is ignored when reporting the cursor position. The position reported is always relative to the origin of the virtual screen.

Note: This report is sent from the client to the host in response to a device status report 6 (DSR 6).

Escape Sequence	**ESC[Pr,PcR**
Modes that affect this code	**None**

where:

Pr is the row position.

Pc is the column position.

R is the final character.

Example

Refer to the following example of a CPR sent from the client to the host:

```
printf("\033[2;5H");        /*CUP-Place cursor*/
printf("\033[6n");          /*DSR-Request cursor position*/
gets(s);                    /*String returned=
                            "\033[2;5R"*/
```

Note: This example requires that the [Enter] key is pressed to exit from the gets *command.*

See Also

DSR

CTC – Cursor Tabulation Control

Function

CTC sets or clears the horizontal tab positions.

> **Escape Sequence**　　　　　**ESC[PnW**
>
> **Modes that affect this code**　　**None**

where:

> **Pn** is the tab position you want to set.
>
> **W** is the final character.

Parameter Values

Refer to the following table for the effect the value that you enter has on the cursor's tab position:

When you enter...	Then this parameter value...
0 (default)	with TSM set, sets the tab stop at the cursor's position.
	with TSM reset, sets the tab stop at the cursor's column, on all rows in the virtual screen.
1	LXE does not support this parameter value.
2	with TSM set, clears the tab stop at the cursor's position.
	with TSM reset, clears the tab stop at the cursor's column, on all rows in the virtual screen.
3	LXE does not support this parameter value.
4	with TSM set, clears all of the tab stops on the cursor's row in the virtual screen.
	with TSM reset, clears all of the tab stops on the virtual screen.
5	with TSM set or reset, clears all of the tab stops on the virtual screen.
6	LXE does not support this parameter value.

Example

Refer to the following example on how a programmer can format a CTC command in the C programming language:

```
printf("\033[5W");              /*clear all horizontal tab
                                positions*/
```

CUB – Cursor Back

Function

CUB moves the cursor one or more positions to the left, up to the left margin. The cursor will stick at the left margin if a large number is entered.

Escape Sequence	**ESC[PnD**
Modes that affect this code	**None**

where:

Pn is the number of character spaces the cursor moves to the left.

D is the final character.

Parameter Values

Refer to the following table for the effect the value that you enter has on the cursor's position:

When you enter...	Then the cursor moves...
0	to the left one space because the value defaults to 1 when you enter 0.
1 (default)	to the left one space.
a number greater than 1	to the left the number of spaces the value indicates up to the left margin.
	Note: Each line contains 80 character positions.

Example

The following example illustrates how a programmer can format a CUB command in the C programming language:

```
printf("\033[4D");              /*CUB-move cursor back 4 spaces*/
```

CUD – Cursor Down

Function

CUD moves the cursor down one or more lines, without changing the cursor's column position.

If the cursor is outside the scrolling region when this command is received, it will not move past the bottom of the virtual screen.

If the cursor is inside the scrolling region, it will not move past the bottom row of the scrolling region.

If Editing Boundary Mode (EBM) is set to the physical display, the scrolling region is ignored and the cursor will not move past the bottom of the physical display.

Note: Scrolling does not occur with this command.

Escape Sequence	**ESC[PnB**
Modes that affect this code	**None**

where:

Pn is the number of lines the cursor moves down.

B is the final character.

Parameter Values

Refer to the following table for the effect the value that you enter has on the cursor's position:

When you enter...	Then the cursor moves...
0	down one line in the same column because the value defaults to 1 when you enter 0.
1 (default)	down one line in the same column.
a number greater than 1	down the number of lines the value indicates without changing column position.

Example

Refer to the following example on how a programmer can format a CUD command in the C programming language:

```
printf("\033[4B");              /*CUD-move cursor down 4 lines*/
```

CUF – Cursor Forward

Function

CUF moves the cursor forward (to the right) one or more spaces up to the right margin. The cursor will stick at the right margin if a large number is entered.

Escape Sequence	**ESC[PnC**
Modes that affect this code	**None**

where:

Pn is the number of character positions the cursor moves to the right.

C is the final character.

Parameter Values

Refer to the following table for the effect the value that you enter has on the cursor's position:

When you enter...	Then the cursor moves...
0	one position to the right because the value defaults to 1 when you enter 0.
1 (default)	one position to the right.
a number greater than 1	to the right the number of times the value indicates up to the right margin.
	Note: Each line contains 80 character positions.

Example

Refer to the following example on how a programmer can format a CUF command in the C programming language:

```
printf("\033[4C");          /*CUF-moves cursor forward 4
                            spaces*/
```

CUP – Cursor Position Absolute

Function

CUP moves the cursor to a specified row/column position.

If the row or column positions are out of range, the client will adjust them and be within range of the appropriate display.

Escape Sequence	**ESC[Pr,PcH**
Modes that affect this code	**None**

where:

Pr is the row position.

Pc is the column position.

H is the final character.

Parameter Values

Refer to the following table for the values that you can enter for CUP:

Parameter	Value	
Pr	1-25	Line 25 is the status line.
Pc	1-80	

Note: If you enter 0 for Pr or Pc, the parameter defaults to 1.

Example

Refer to the following example on how a programmer can format a CUP command in the C programming language:

```
printf("\033[3;17H");          /*CUP-move cursor to row 3, column
                               17*/
```

CUU – Cursor Up

Function

CUU moves the cursor up one or more lines, without changing the cursor's column position.

Note: *Scrolling does not occur with this command.*

Escape Sequence	**ESC[PnA**
Modes that affect this code	**None**

where:

Pn is the number of lines the cursor moves up.

A is the final character.

Parameter Values

Refer to the following table for the effect the value that you enter has on the cursor's position:

When you enter...	Then the cursor moves...
0	up one line because the value defaults to 1 when you enter 0.
1 (default)	up one line.
a number greater than 1	up the number of lines the value indicates up to the top of the virtual screen.

Example

Refer to the following example on how a programmer can format a CUU command in the C programming language:

```
printf("\033[4A");            /*CUU-move cursor up 4 lines*/
```

CVT – Cursor Vertical Tab

Function

CVT is processed the same as the Cursor Next Line command (CNL).

Escape Sequence	**ESC[PnY**
Modes that affect this code	**None**

where:

Pn is the number of lines the cursor moves down.

Y is the final character.

See

CNL for more details.

DAQ – Define Area Qualifications

Function

DAQ indicates that the cursor's position is the start of a string of consecutive character positions that constitute a qualified area. The end of a qualified area is indicated by the beginning of the next qualified area. A qualified area restricts the type of input the client user enters.

Escape Sequence	**ESC[Pno**
Modes that affect this code	**None**

where:

Pn indicates the type of data that the client accepts.

o is the final character.

Multiple parameters may be sent in a single DAQ command if they are separated by semi-colons.

Parameter Values

Refer to the following table for the effect the value that you enter has on the qualified area:

When you enter...	Then the qualified area...
0 (default)	accepts all data.
1	does not accept any input and is not selected to transmit.
	Note: Entering 1 for this value is equivalent to the 8-bit control code start of protected area (SPA) discussed in C1 Codes: 8-Bit Control Codes.
2	accepts only displayable characters, 20-7E hex.
3	accepts only numeric characters.
4	accepts only alphabetic characters.
5	right justifies the data.
6	is filled with zeros.
7	has a horizontal tab stop set at the start of area.
8	does not accept any input, but it is selected to transmit.
9	is filled with spaces.
10	accepts input, but the characters are invisible.
15	accepts only keyboard input.

Every new DAQ command that the client receives will always overwrite the existing DAQ attribute type from the cursor to the end of the area.

Note: Parameter types 6 and 9 are executed immediately and are not stored in the TE.

Rule

All DAQ parameters are supported in block mode. Parameters 0, 2, 3, 4 and 10 are supported in scroll mode. No parameters are supported in character mode.

However, if multiple DAQ parameters are specified in a single command, they will combine as described in the table below.

Parameter in DAQ command	+	Previous Parameter in DAQ command	=	Parameter Stored in TE
All data (0)	+	any	=	All data
Guarded (1)	+	any	=	Guarded
Displayable (2)	+	any	=	Displayable
Numeric (3)	+	Alphabetic	=	Alphanumeric
Numeric (3)	+	any other	=	Numeric
Alphabetic (4)	+	Numeric	=	Alphanumeric
Alphabetic (4)	+	any other	=	Alphabetic
Right Justify (5)	+	any	=	Previous+Right Justified
				(If 5 is the only parameter received, the area will be an "All data, right justified" one.)
Regular Tab (7)	+	any	=	All data+Regular Tab Stop
				(Ignores TSM)
Selected (8)	+	any	=	Selected

For example, if the host specified ESC[1;5;0o, the client would mark the area as simply "All data" since 0 was the last parameter. However, if the host specified ESC[3;4;5o, the client would mark the area as alpha-numeric and right justified.

Example

Refer to the following example on how a programmer can format a DAQ command in the C programming language:

```
printf("\033[1o");          /start of a protected area*/
printf("Accept numeric character input only.\n");
printf("\033[3o");          /*Define numeric entry area start*/
printf("____");             /*4 underscore characters*/
printf("\033[1o");          /*define numeric entry area stop*/
printf("\033[4D");          /*CUB-moves the cursor back 4
                            spaces*/
gets(s);                    /*string returned will be numeric
                            value only*/
```

DCH – Delete Character

Function

DCH deletes the character(s) at the cursor position and shifts all the characters right of the cursor to the left (toward the cursor).

The client will not delete past the first protected screen cell to the right of the cursor. If the cursor is on a protected cell when the DCH command is received, it will move one position to the right (i.e. CUF 1). No deletion takes place in this case.

Note: The vacated character position at the other end is erased.

Rule

Block mode must be enabled to use DCH.

Escape Sequence	**ESC[PnP**
Modes that affect this code	**SEM**

where:

Pn is the number of characters to delete.

P is the final character.

Parameter Values

Refer to the following table for the effect the value that you enter has on the cursor's position:

When you enter...	Then the client...
0	deletes one character because the value defaults to 1 when you enter 0.
1 (default)	deletes one character.
a number greater than 1	deletes the number of characters the value indicates.

Example

Refer to the following example on how a programmer can format a DCH command in the C programming language:

```
printf("All X's on this XX line will be deleted");
printf("\033[23D");          /*CUB-23 spaces back*/
printf("\033[2P");           /*DCH-deletes 2 x's*/
printf("\033[12D");          /*CUB-12 spaces back*/
printf("\033[1P");           /*DCH-deletes x's remaining*/
```

DL – Delete Line

Function

DL deletes the line the cursor is in and shifts all following lines up. If the cursor is outside the scrolling region, this command is ignored.

Escape Sequence	**ESC[PnM**
Modes that affect this code	**None**

where:

Pn is the number of lines the device deletes.
M is the final character.

Parameter Values

Refer to the following table for the number of lines the **client** deletes according to the value that you enter:

When you enter...	Then the client deletes...
0	one character because the value defaults to 1 when you enter 0.
1 (default)	one line.
a number greater than 1	the number of lines the value indicates without going beyond the bottom of the virtual screen or the scrolling region.

Example

Refer to the following example on how a programmer can format a DL command in the C programming language:

```
printf("This line will be deleted.\n");
printf("This line will be deleted.\n");
printf("This line moves up and is not deleted.\n");
printf("\033[3A");          /*CUU-cursor moves up 3 lines*/
printf("\033[2M");          /*DL-2 lines are deleted and
                              remaining lines move up*/
```

DMI – Disable Manual Input

Function

DMI disables keyboard and RS-232 input. This command is equivalent to SM KAM.

Escape Sequence	**ESC`** (tilde)
Modes that affect this code	**None**

Example

Refer to the following example on how a programmer can format a DMI command in the C programming language:

```
printf("\033`");            /* Keyboard and RS-232 now locked
                            */
```

DSR – Device Status Report

Function

DSR requests information or reports information. DSR from the host requests information from the client receiving the sequence (Pn = 5, 6). DSR from the client reports the status of the client (Pn = 0).

Escape Sequence	**ESC[Pnn**
Modes that affect this code	**None**

where:

Pn is either a value that you enter (5 or 6) or it is a value returned from the client (0).

n is the final character.

Parameters Indicating a Request

Refer to the following table for the requests the host sends to a client according to the parameter value:

Note: Parameter 5 and 6 do not append a CR or LF to the response. Parameters 90, 92 and 93 append CR and LF as if the [SEND] key was pressed.

Parameter	Report that the Host is Requesting
5	Status Report. *Note: The client returns a CSI0n indicating ready status.*
6	Cursor Position Report (CPR).
90	Sends the TE display, just as if the *client* user had pressed the [Enter] key.
92	Sends the TE display just as if the *client* user had pressed [Enter].
93	Same as 90, except the TE does not lock its keyboard.
94	Sends the upper left and lower right coordinates of the physical display in the format `CSI?UL Row;UL Col;LR Row,LR Col R` The coordinates are 1-based.

Note: DSR 90 - 94 are useful for debugging host applications.

Response to a CPR Request

The *client* sends a CPR response when the host requests a CPR. Refer to *CPR* in this section for more information on the cursor position report.

Note: *A carriage return (CR) and a line feed (LF) are not returned in the response to a CPR request.*

Example

Refer to the following example on how a programmer can format a DSR command that requests a *client* to send a status report in C programming language:

```
printf("\033[5n");
gets (s);                    /*string returned is \033[0n for a
                             ready status*/
```

Note: *The example above requires that the [Enter] key be pressed at the* client *to exit from the* gets *command.*

EA – Erase in Area

Function

EA erases characters in the qualified area where the active position (cursor's position) resides according to the value that you enter. This command will <u>not</u> erase the type of qualified area that the cursor is in.

Escape Sequence	**ESC[PnO**
Modes that affect this code	**ERM**

where:

Pn indicates what portion of the area is erased.
O is the final character.

Parameter Value

Refer to the following table on how characters are erased in the qualified area:

When you enter...	Then the client ...
0 (default)	erases characters from the cursor to the end of the qualified area.
1	erases characters from the start of the qualified area to the cursor.
2	erases all characters in the qualified area.

Example

Refer to the following example on how a programmer can format an EA command in the C programming language:

```
printf("\033[6l");           /*Reset ERM*/
printf("\033[1o")            /*DAQ-Start of protected area*/
printf("This will not be erased.\n");
printf("\033[0o");           /*DAQ-Accepts all input*/
printf("_ _ _ _ \033[1o");  /*DAQ-Accept no input*/
printf("\033[4D");           /*CUB-4 spaces*/
printf("\033[2O");           /*EA-Erase in Area-all*/
```

ECH – Erase Character

Function

ECH erases the character at the cursor's position and possibly other characters to the right of the cursor, depending on the value that you enter. If a large number of characters to erase is specified, the erase will continue on succeeding lines.

Note: The active (cursor) position is unchanged.

Escape Sequence	**ESC[PnX**
Modes that affect this code	**ERM**

where:

Pn is the number of characters to erase.
X is the final character.

Parameter Values

Refer to the following table on how the client erases characters according to the value that you enter:

When you enter...	Then the client ...
0	erases one character because the value defaults to 1 when you enter 0.
1 (default)	erases one character.
a number greater than 1	erases the number of characters the value indicates.

Example

Refer to the following example on how a programmer can format an ECH command in the C programming language:

```
printf("This Z will be erased.");
printf("\015");              /*Carriage return*/
printf("\033[5c");           /*Cursor forward 5 spaces*/
printf("\033[1X");           /*Erase character Z*/
```

ED – Erase in Display

Function

ED erases some or all of the characters in the display according to the value that you enter. This command will erase the type of qualified area that the cursor is in. It will also erase qualified area boundaries.

If EBM is set to the virtual screen, every row that is completely erased will have its line attribute reset to single high and single wide.

Escape Sequence	**ESC[PnJ**
Modes that affect this code	**ERM**

where:

Pn indicates what portion of the display is erased.

J is the final character.

Parameter Values

Refer to the following table on how the client erases characters according to the value that you enter:

When you enter...	Then the client ...
0 (default)	erases characters starting with the cursor position to the end of the display.
1	erases the characters from the start of the display to the cursor.
2	erases all the characters on the display. The cursor is moved to the origin of the physical display or virtual screen, depending on the EBM setting.

Example

Refer to the following example on how a programmer can format an ED command in the C programming language:

```
printf("\033[2J");            /*Erases entire display*/
```

EF – Erase in Field

Function

EF erases characters in the current field depending on the value that you enter. This command will <u>not</u> erase the type of qualified area that the cursor is in.

Escape Sequence	**ESC[PnN**
Modes that affect this code	**EBM**

where:

Pn indicates what portion of the current field is erased.

N is the final character.

Parameter Values

Refer to the following table on how the client erases characters in a field according to the value that you enter:

When you enter...	Then the client ...
0 (default)	erases characters from the cursor position to the end of the field.
1	erases characters from the beginning of the field to the cursor position.
2	erases the entire field that the cursor is in.

Example

Refer to the following example on how a programmer can format an EF command in the C programming language:

```
printf("\033[5W");          /*CTC-Clear all tabs*/
printf("\033[0W");          /*CTC-Set a tab*/
printf("_ _ _ _ \033[0W");  /*CTC-Set a tab*/
printf("\033[3D");          /*CUB-3 spaces*/
printf("\033[2N");          /*EF-Erases all of field*/
```

EL – Erase in Line

Function

EL erases characters in the line the cursor is in according to the value that you enter. This command <u>will</u> erase the type of qualified area that the cursor is in. It will also erase qualified area boundaries.

Escape Sequence	**ESC[PnK**
Modes that affect this code	**ERM**

where:

Pn indicates what portion of the current line is erased.

K is the final character.

Parameter Values

Refer to the following table on how the client erases characters according to the value that you enter:

When you enter...	Then the client ...
0 (default)	erases characters from the cursor position to the end of line.
1	erases characters from the beginning of the line to the cursor.
2	erases the entire line that the cursor is in.

Example

Refer to the following example on how a programmer can format an EL command in the C programming language:

```
printf("\033[6l");          /*Reset ERM*/
printf("\033[1o");          /*DAQ-Start of protected area*/
printf("This will not be erased.\n");
printf("\033[0o");          /*DAQ-Accepts all input*/
printf("_ _ _ _ \033[1o");  /*DAQ-Accepts no input*/
printf("\033[4D");          /*CUB-4 spaces*/
printf("\033[2K");          /*Erase in Line*/
```

EMI – Enable Manual Input

Function

EMI enables keyboard and RS-232 input. This command is equivalent to RM KAM.

Escape Sequence	**ESCb**
Modes that affect this code	**None**

Example

Refer to the following example on how a programmer can format an EMI command in the C programming language:

```
printf("\033b");            /* Keyboard and RS-232 now unlocked
                            */
```

HVP – Horizontal Vertical Position

Function

HVP moves the cursor to the specified row/column position. HVP is the same as the CUP command in LXE Legacy clients. Refer to the CUP command for more details.

Escape Sequence	**ESC[Pr,Pcf**
Modes that affect this code	**None**

where:

Pr is the row position.

Pc is the column position.

f is the final character.

Parameter Values

Refer to the following table for the values that you can enter for HVP:

Parameter	Value	
Pr	1 - 25	(Line 25 is the status line.)
Pc	1 - 80	

Note: If you enter 0 for Pr or Pc, the parameter defaults to 1.

Example

Refer to the following example on how a programmer can format an HVP command in the C programming language:

```
printf("\033[3;17f");       /*Move the cursor to row 3, column
                            17*/
```

ICH – Insert Character

Function

ICH shifts all characters from the cursor to the right and inserts a space character at the cursor position.

The client will not insert past the first protected screen cell to the right of the cursor. If the cursor is on a protected cell when this command is received, it will move one position to the right (i.e. CUF 1). No insertion takes place in this case.

Escape Sequence	**ESC[Pn@**
Modes that affect this code	**SEM**

where:

Pn is the number of character spaces the cursor shifts to the right.

@ is the final character.

Rule

Block mode must be enabled for ICH to work.

Parameter Values

Refer to the table below on how the characters move according to the value that you enter:

When you enter...	Then the client shifts the characters...
0	one space to the right because the value defaults to 1 when you enter 0.
1 (default)	one space to the right.
a number greater than 1	the number of spaces the value indicates to the right.

Example

Refer to the following example on how a programmer can format an ICH command in the C programming language:

```
printf("Insert space here");
printf("\015");                 /*CR-carriage return*/
printf("\033[6C");              /*CUF-cursor forward*/
printf("\033[1@");              /*ICH-insert space character*/
printf("\033[1C");              /*CUF-cursor forward*/
printf("\033[1@");              /*ICH-insert space character*/
```

IL – Insert Line

Function

IL inserts one or more blank lines at the cursor's position and shifts all following lines down. If the cursor is outside the scrolling region, this command is ignored.

Escape Sequence	**ESC[PnL**
Modes that affect this code	**None**

where:

Pn is the number of blank lines inserted at the cursor's position.

L is the final character.

Parameter Values

Refer to the following table on how to insert blank lines at the cursor's position:

When you enter...	Then the client inserts...
0	one blank line because the value defaults to 1 when you enter 0.
1 (default)	one blank line.
a number greater than 1	the number of blank lines the value indicates at the cursor's position. The inserted lines do not go beyond the bottom of the virtual screen or the scrolling region.

Example

Refer to the following example on how a programmer can format an IL command in the C programming language:

```
printf("This line will move down 2 lines.");
printf("\015");              /*CR-carriage return*/
printf("\033[2L");           /*Insert 2 lines*/
```

MC – Media Copy

Function

MC enables or disables the RS-232 port. When enabled, the client sends received characters to the RS-232 port without displaying them on the screen. All characters (except: Nul, XON, XOFF, DC4, DC2, CSI5i and CSI4i) are sent to the port. When disabled, the client sends received chars to the display. MC has a higher priority than autoprint mode.

Note: An alternative way to disable MC is to send a DC4.

Escape Sequence	**ESC[Pni**
Modes that affect this code	**None**

where:

Pn indicates whether the RS-232 port is off or on.

i is the final character.

Parameter Values

Refer to the table below on how to enable or disable the RS-232 port:

When you enter...	Then the RS-232 port is ...
4	off.
5	on.

Example

Refer to the following example on how a programmer can format an MC command in the C programming language:

```
printf("\033[5i"); /*Enables following text to copy to the
RS-232 port*/
printf("This message will copy to the RS-232 port.");
printf("\033[4i"); /*Disables RS-232 port*/
```

REP – Repeat Character

Function

REP repeats the last character transmitted by the host the number of times indicated by the value. Only characters are repeated. Escape sequences are not repeated.

Escape Sequence	**ESC[Pnb**
Modes that affect this code	**None**

where:

Pn is the number of times a character is repeated.

b is the final character.

Parameter Values

Refer to the table below on how many times the character is repeated according to the value that you enter:

When you enter...	Then the client repeats the character ...
0	once because the value defaults to 1 when you enter 0.
1 (default)	once.
a number greater than 1	the number of time that the value indicates.

Example

Refer to the following example on how a programmer can format an REP command in the C programming language:

```
printf("The number 3 is repeated:\n);
printf("4 times-3");
printf("\033[4b");          /*repeat 4 times*/
```

RM – Reset Mode

Function

RM resets one or more modes for the device indicated by the value that you enter. Several modes can be reset with the same command if the parameter values are separated by semicolons. They can be sent in any order except #3.

Escape Sequence	**ESC[Pn1**
Modes that affect this code	**None**

where:

Pn indicates which mode to reset.

l (lower-case L) is the final character.

Parameter Values

Refer to the following table for the modes RM resets:

Parameter	Mode to Reset	Description
1	Guarded Area Transfer Mode (GATM)	Enables the client to transmit only the unguarded areas to the host and auxiliary device. See the "Sending the Client Screen to the Host" section for more details.
2	Keyboard Action Mode (KAM)	Unlocks the keyboard and RS-232 data entry.
4	Insert-Replace Mode (IRM)	Enables the client to replace the information during editing by overwriting the content of the line.
6	Erasure Mode (ERM)	Determines that the editing commands such as EA cannot erase protected characters.
12	Send-Receive Mode (SRM)	Enables local echo in the client so that the client operates in scroll or block mode. Character mode communication is disabled.
15	Multiple Area Transmit Mode (MATM)	Enables the client to transmit only the selected area containing the cursor. See the "Sending the Client Screen to the Host" section for more details.
17	Selected Area Transmit Mode (SATM)	Enables the client to transmit only the selected areas. See the "Sending the Client Screen to the Host" section for more details.
18	Tab Stop Mode (TSM)	Determines that the tab stop settings affect the entire virtual display.
20	Line Feed/New Line Mode (LNM)	Enables the client to perform only a line feed while the cursor remains in its current column.

Example

Refer to the following example on how a programmer can format an RM command in the C programming language:

```
printf("\033[4l")              /*reset IRM*/
```

See Also

Refer to *SM* in this section for more information on how to set the modes discussed in the previous table.

SEM – Select Editing Extent Mode

Function

SEM indicates the extent to which the delete character (DCH) command and the insert character (ICH) command affect the display window.

Escape Sequence	**ESC[PnQ**
Modes that affect this code	**None**

where:

Pn indicates how much of the display window is affected by the editing commands.

Q is the final character.

Parameter Values

Refer to the following table for the affect that SEM has on the DCH and ICH editing commands:

When you enter...	Then DCH and ICH commands affect...
0 (default)	the entire display.
1	only the row the cursor is in.
2	only the field containing the cursor (between horizontal tab stops).
3	only the selected area containing the cursor (defined by DAQ, SPA and EPA).

Example

Refer to the following example on how a programmer can format an SEM command in the C programming language:

```
printf("\033[3Q");              /*SEM-3*/
```

See Also

Refer to the "DAQ" section in this chapter for more information on define area qualification. Refer to the "EPA" and "SPA" sections of Chapter 5 "Control Codes" for more information on end of protected area and select protected area.

SM – Set Mode

Function

SM sets one or more modes for the client as specified by the parameter. Several modes can be set with the same command if the parameter values are separated by semicolons. They can be sent in any order except #3.

Escape Sequence	**ESC[Pnh**
Modes that affect this code	**None**

where:

Pn indicates the mode to set.

h is the final character.

Parameter Values

Refer to the following table for the modes SM sets:

Parameter	Mode to Set	Description
1	Guarded Area Transfer Mode (GATM)	Enables the client to transmit the entire screen to the host or auxiliary device. See the "Sending the Client Screen to the Host" section for more details.
2	Keyboard Action Mode (KAM)	Locks the keyboard and RS-232 data entry. The keyboard can not be unlocked from the client.
3	Control Representation Mode (CRM)	Enables the client to display the control codes. This parameter value must immediately precede the final character.
4	Insert-Replace Mode (IRM)	Enables the client to insert the information while the contents of the line shift to the right.
6	Erasure Mode (ERM)	Determines that the editing commands such as EA can erase protected characters.
12	Send-Receive Mode (SRM)	Disables local echo in the client so that the client operates in character mode. Scroll or block mode communication is disabled.
15	Multiple Area Transmit Mode (MATM)	Enables the client to transmit all selected areas. See the "Sending the Client Screen to the Host" section for more details.
17	Selected Area Transmit Mode (SATM)	Enables the client to transmit the entire virtual screen. See the "Sending the Client Screen to the Host" section for more details.

Parameter	Mode to Set	Description
18	Tab Stop Mode (TSM)	Determines that the tab stop settings only affect the cursor's row.
20	Line Feed/New Line Mode (LNM)	Enables the client to perform a line feed and carriage return, placing the cursor at the beginning of the next line.

Example

Refer to the following example on how a programmer can format an SM command in the C programming language:

```
printf("\033[4h");              /*Set IRM*/
```

See Also

RM (for more information on how to reset the modes discussed in the previous table).

TBC – Tab Clear

Function

TBC clears one or more tab stops according to the value that you enter.

Escape Sequence	**ESC[Png**
Modes that affect this code	**None**

where:

Pn is the type of tab stops to clear.

g is the final character.

Parameter Values

Refer to the following table on how TBC affects the tab stop positions:

When you enter...	Then the parameter value ...
0 (default)	with TSM set, clears the tab stop at the cursor's position.
	with TSM reset, clears the tab stop at the cursor's column on all rows in the virtual screen.
1	LXE does not support this parameter.
2	with TSM set, clears all of the tab stops on the cursor's row in the virtual screen.
3	with TSM set or reset, clears all of the tab stops on the virtual screen.
4	LXE does not support this parameter.

Example

Refer to the following example on how a programmer can format a TBC command in the C programming language:

```
printf("\033[0g");          /*Clear tab stop*/
```

DEC Private Escape Sequences

This chapter provides information for the programmer on how to code DEC private escape sequences for LXE equipment. All the private escape sequences supported by LXE are listed in this chapter. If a DEC private escape sequence is not listed, LXE does not support it, and it is treated as a no operation (NOP).

APM – Auto Print Mode

Function

APM turns auto print mode on or off. When auto print is on, the virtual screen row that the cursor is presently on prints when the cursor is moved off the line with a line feed, form feed, vertical tab or auto wrap. The printed line ends with a carriage return and a line feed. Trailing spaces on a screen row will not print.

Escape Sequence	**ESC[?Pni**
Modes that affect this code	**None**

where:

Pn is the parameter value that indicates on or off.
i is the final character.

Parameter Values

Refer to the following table on the parameter values indicating the mode to set:

When you enter...	Auto Print mode is turned ...
4 (default)	off.
5	on.

DECRM – DEC Reset Mode

Function

DECRM resets the mode you choose.

Escape Sequence	**ESC[?Pnl**
Modes that affect this code	**None**

where:

> **?** is a constant.
>
> **Pn** is the value of the mode to reset.
>
> **l** (lowercase L) is the final character.

Parameter Values

Refer to the following table for the modes DECRM can reset:

Param.	Mode to Reset	Description
18	Print Form Feed Mode (DECPFF)	Disables the terminating character (form feed) after print screen operations. (Print full screen or scroll region with the PS command.)
19	Print Extent Mode (DECPEX)	Selects the scrolling region to print for each Print Screen (PS) command issued.

DECSCA – Select Character Attribute

Function

DECSCA indicates the start of a protected or unprotected string of character positions.

Escape Sequence	**ESC[Pn"q**
Modes that affect this code	**None**

where:

Pn is the parameter value that indicates the character attribute.

" and **q** are the final characters.

Parameter Values

Refer to the following table on the parameter values indicating the character attribute as protected or unprotected:

When you enter...	Then the character(s) is ...		
0 (default)	unprotected.	*Note:*	*This is the same as the EPA command.*
1	protected.	*Note:*	*This is the same as the SPA command.*
2	unprotected.	*Note:*	*This parameter has the same effect as parameter 0.*

Example

Refer to the example below on how a programmer can format a DECSCA command in the C programming language:

```
printf("\033[1\"q");        /*DECSCA-1 cannot erase*/
printf("This text is protected from erasure.\n");
printf("\033[0\"q");        /*DECSCA-0 can erase*/
printf("This text is not protected.");
```

DECSED – Selective Erase in Display

Function

DECSED enables the programmer to erase some or all of the erasable characters in the display. This command will not erase qualified area boundaries or the type of qualified area the cursor is in.

If EBM is set to the virtual screen, every row that is completely erased will have its line attribute reset to single high and single wide.

Escape Sequence	**ESC[?PnJ**
Modes that affect this code	**None**

where:

? is a constant.

Pn is the portion of the display to erase.

J is the final character.

Parameter Values

Refer to the following table on how the characters are erased according to the value that you enter:

When you enter...	Then the characters are erased from ...
0 (default)	the cursor through the end of the display.
1	the beginning of the display through the cursor.
2	the entire display. The cursor is moved to the origin of the physical or virtual display depending on the EBM setting.

Example

Refer to the example below on how a programmer can format a DECSED command in the C programming language:

```
printf("\033[0\"q");          /*DECSCA = 0*/
printf("Text from this letter x to the end of \n");
printf("the display should be erased.\n");
printf("\033[2A");            /*CUU-Cursor up 2 lines*/
printf("\033[22C");           /*CUF-Cursor Forward 22 to the x*/
printf("\033[?0J");           /*Erase from x to the end of the
                               display*/
```

DECSEL – Selective Erase in Line

Function

DECSEL enables the programmer to erase some or all of the erasable characters in a single line of text. This command will not erase qualified area boundaries or the type of qualified area the cursor is in.

Escape Sequence	**ESC[?PnK**
Modes that affect this code	**None**

where:

? is a constant.

Pn is the portion of the line to erase.

K is the final character.

Parameter Values

Refer to the following table on how the characters are erased in relation to the cursor:

When you enter...	Then the characters are erased from ...
0 (default)	the cursor through the end of the line.
1	the beginning of the line through the cursor.
2	the entire line.

Example

Refer to the example below on how a programmer can format a DECSEL command in the C programming language:

```
printf("\033[0\"q");          /*DECSCA = 0*/
printf("Text from this letter x to the end of \n");
printf("the above line should be erased.\n");
printf("\033[2A");            /*CUU-Cursor up 2 lines*/
printf("\033[22C");           /*CUF-Cursor forward 22*/
printf("\033[0K");            /*Erase from x to the end of the
                                line*/
```

DECSM – DEC Set Mode

Function

DECSM sets the mode you choose.

Escape Sequence	**ESC[?Pnh**
Modes that affect this code	**None**

where:

? is a constant.

Pn is the value of the mode to set.

h is the final character.

Parameter Values

Refer to the following table for the modes that DECSM can set:

Parameter	Mode to Reset	Description
18	Print Form Feed Mode (DECPFF)	Enables the terminating character (form feed) after print screen operations. (Print full screen with the PS command.)
19	Print Extent Mode (DECPEX)	Selects the full screen to print for each Print Screen (PS) command issued.

DECDHL – Double-Height Line

Function

Makes the line with the cursor on it the top or bottom half of a double-height, double-width line. Pairs of this command must be used on adjacent lines.

The same screen character must be used on both the top and bottom lines in that column to form a full character.

Note: When this command is received, the device will reset the right half of the virtual screen to spaces and remove any tabs or areas contained there. If the cursor is in the right half, it will be moved to the last column in the left half.

Top Half

Escape Sequence	**ESC#3**
Modes that affect this code	**None**

Bottom Half

Escape Sequence	**ESC#4**
Modes that affect this code	**None**

Example

Refer to the following example on how a programmer can format a DECDHL command in the C programming language:

```
printf("\033#3 Double High Line\n");
printf("\033#4 Double High Line\n");
```

DECDWL – Double-Width Line

Function

Makes the line with the cursor on it double-width, single-height.

Note: When this command is received, the device will reset the right half of the virtual screen to spaces and remove any tabs or areas contained there. If the cursor is in the right half, it will be moved to the last column in the left half.

Escape Sequence	**ESC#6**
Modes that affect this code	**None**

Example

Refer to the following example on how a programmer can format a DECDWL command in the C programming language:

```
printf("\033#6 Double-Width Line\n");
```

G0/G1/G2/G3 – Designate G0/G1/G2/G3

Function

G0 designates one of the hard character sets (ASCII, DEC Supplemental Graphics, DEC Special Graphics, and National Replacement character sets) as G0 through G3.

Escape Sequence	To Designate
ESC(F	G0
ESC)F	G1
ESC*F	G2
ESC+F	G3

where:

F is the final character from the following table.

Character set to designate...	Final character...
ASCII	B
DEC Supplemental	<
DEC Special Graphics	0
National Replacement Character Sets	various final characters dependent upon mobile device operating system and device specific settings

Refer to Appendix B "ANSI Quick Reference Guide" for the **DEC Special Graphics Code Table**.

PL – Print Line

Function

PL prints the display line containing the cursor. A carriage return and line feed are automatically printed after the line.

Escape Sequence	**ESC[?1i**
Modes that affect this code	**None**

Example

Refer to the following example on how a programmer can format a PL command in the C programming language:

```
printf("\033[?1i");    /*Prints the line the cursor is on*/
```

PS – Print Screen

Function

PS prints the display to the RS-232 port. A carriage return and line feed is automatically printed after each line.

Escape Sequence	**ESC[Pni**
Modes that affect this code	**DECPEX, DECPFF**

where

> **Pn** is either 0 or omitted.

Example

Refer to the following example on how a programmer can format a PS command in the C programming language:

```
printf("\033[0i");    /*Causes the display to be printed*/
```

S7C1T – Select 7-Bit C1 Control Transmission

Function

S7C1T informs the client to convert all C1 codes to their 7-bit equivalent before sending them back to the host.

Note: The client will always accept either 7-bit or 8-bit control codes regardless of the state of this command.

Escape Sequence	**ESC sp F**
Modes that affect this code	None

Example

Refer to the following example on how a programmer can format a S7C1T command in the C programming language:

```
printf("\033 F");          /*There is a single space between
                           the 3 and the F*/
```

S8C1T – Select 8-Bit C1 Control Transmission

Function

S8C1T informs the client to send all C1 codes back to the host without converting them to their 7-bit equivalent.

Note: The client will always accept either 7-bit or 8-bit control codes regardless of the state of this command.

Escape Sequence	**ESC sp G**
Modes that affect this code	None

Example

Refer to the following example on how a programmer can format a S8C1T command in the C programming language:

```
printf("\033 G");          /*There is a single space between
                           the 3 and the G*/
```

SI – Shift In (Lock Shift G0)

Function

Invokes G0 character set into GL.

Hexadecimal Value	Control Code
0F	^O

Example

Refer to the following example on how a programmer can format an SI command in the C programming language:

```
printf("\033(0"); /* Designate DEC special graphics into
                     G0. */
printf("\017");   /* Shift G0 into GL */
printf("lqqk\n"); /* These two lines print a small box on
                     the display. */
printf("mqqj");
```

SO – Shift Out (Lock Shift G1)

Function

Invokes G1 character set into GL.

Hexadecimal Value	Control Code
0E	^N

Example

Refer to the following example on how a programmer can format an SO command in the C programming language:

```
printf("\033)0"); /* Designate DEC special graphics into
                     G1. */
printf("\016");   /* Shift G1 into GL */
printf("lqqk\n"); /* These two lines print a small box on
                     the display. */
printf("mqqj");
```

LXE Private Escape Sequences

LXE offers special escape sequences as well as command strings that modify terminal emulation functions, window parameters, and wild card local edit characters. LXE also provides command strings for stored forms, using a backspace function with the [CLEAR] key, enabling an echo suppress, and a wide range of beep patterns.

This chapter describes the applications for each command string and information on how the programmer can code the escape sequences and command strings.

LXERM – Reset Mode

Function

LXERM resets the mode that you choose.

Escape Sequence	**ESC[=Pnl**
Modes that affect this code	**None**

where:

= is a constant.

Pn is the value of the mode to reset.

l (lowercase L) is the final character.

Parameter Values

Refer to the following table for the modes LXERM can reset:

Param.	Mode to Reset	Description
1	Scroll/Block Mode (LXELM)	Enables block mode. Rule: Send-receive mode (SRM) must be reset for LXELM to work.
2	New Line Echo Mode (LXENE)	Disables new line local echo. If the client is in scroll/block mode, does not execute carriage return and line feed to the client display when the [Enter] key is pressed.
3	Autosend (LXEAS)	Disables autosend. In other words, the client transmits only when the [Enter] key is pressed. Rule: Send-receive mode (SRM) must be reset for LXEAS to work.

Example

Refer to the following example on how a programmer can format an LXERM command in the C programming language:

```
printf("\033[=3l");          /*Reset LXEAS- Reset autosend
                             mode*/
```

LXESM – Set Mode

Function

LXESM sets the mode you choose.

Escape Sequence	**ESC=Pnh**
Modes that affect this code	**None**

where:

= is a constant.

Pn is the value of the mode to set.

h is the final letter.

Parameter Values

Refer to the following table for the modes LXESM can set:

Parameter	Mode to Set	Description
1	Scroll/Block Mode (LXELM)	Enables scroll mode. Rule: Send-receive mode (SRM) must be reset for LXELM to work.
2	New Line Echo Mode (LXENE)	Enables new line local echo. If the client is in scroll/block mode, execute carriage return and line feed to the client display when the [Enter] key is pressed.
3	Autosend (LXEAS)	Enables autosend. In other words, the client transmits the screen when the last field is filled with a bar code entry, keyboard entry or RS-232 port entry. Rule: Send-receive mode (SRM) must be reset for LXEAS to work.

Example

Refer to the following example on how a programmer can format an LXESM command in the C programming language:

```
printf("\033[=3h");          /*Set LXEAS-Set autosend mode*/
```

Specifying Bar Code, Keypad and Control Modes

This section discusses the elements of the command string that enable the programmer to set the bar code, keypad and control modes for the client. The different settings for each parameter of the command string are discussed in detail.

Command String for Bar Code, Keypad and Control Modes

The programmer can specify bar code, keypad and control modes with the appropriate command string. This section discusses elements of the command string and its structure.

Command String

Use the following command string for specifying bar code, keypad, and control modes:

```
CSIc1;p1;p2;p3;p4;p5;p6;p7;p8;p9;p10+E
```

Rule

Each parameter must be preceded by a semicolon (;).

You can use the semicolon as a placeholder if you want to change only one parameter. For example, you can send the following string to change only p5 of the command string:

```
CSI1;;;;;1+E
```

String Elements

The following table describes the different elements of the command string for specifying bar code, keypad, and control modes:

String Element	Description
CSI	ANSI control sequence introducer (Hex 9B for 8-bit code or Hex 1B5B for 7-bit code).
c1	A compatibility byte that ensures the terminal emulation understands the command string sent to it. The c1 is a revision level character for compatibility purposes, which increments with each revision. The original value for this parameter is 1.
;	A required delimiter that must precede all parameter entries.
p2	Bar Code/RS-232 Input Left Justify Mode 0 = Left Justify Mode Off 1 = Left Justify Mode On The section "Left Justify Mode: p2" found in this chapter discusses this parameter in more detail.

String Element	Description
p3	Bar Code/RS-232 Input Clear Mode 0 = Clear Off 1 = Clear On The section "Clear Mode: p3" found in this chapter discusses this parameter in more detail.
p4	Bar Code/RS-232 Input Truncate Mode 0 = Truncate OFF 1 = Truncate ON The section "Truncate Mode: p4" found in this chapter discusses this parameter in more detail.
p5	Bar Code/RS-232 Input Advance Mode 0 = Advance OFF 1 = Advance ON The section "Advance Mode: p5" found in this chapter discusses this parameter in more detail.
+E	The final characters of the ANSI command string. This combination indicates the end of the command string and identifies the command string as a bar code, keypad, and control mode command.

Left Justify Mode: p2

Left justify mode determines where the cursor appears in a field from one bar code entry to the next. The programmer must decide if the cursor in a field justifies to the left each time a new bar code entry occurs or if the bar code entry appears at the end of the previous entry.

Rule

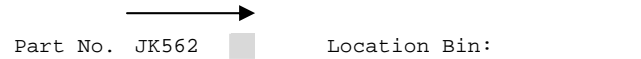You must enable block mode to use left justify mode.

Parameter Values

Refer to the following table on how to set p2 of the command string to disable or enable left justify mode:

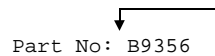| When you enter ...	Then left justify is ...
0	off.
1 (default)	on.

Left Justify Off

Left Justify Off enables a bar code read to appear in the next available cursor position of a field directly following a previous bar code entry. For example, the first bar code entry fills the first five positions of the first field available as shown below:


```
Part No.  JK562           Location Bin:    _   _   _
```

The next bar code read is entered starting at the cursor's position and proceeding to the next unprotected field.

Left Justify On

Left Justify On enables the cursor to left justify in a field each time a bar code read occurs. For example, the following bar code read appears in the field and then the cursor left justifies when the next bar code entry occurs. The next bar code read appears in the field starting at the B position:


```
Part No: B9356
```

Clear Mode: p3

Clear mode determines if a field is automatically cleared when the next bar code entry occurs or if the data in the field is overwritten by the new entry.

IMPORTANT   If a field is not cleared after the data is transmitted to the host, there is a chance that some of the previous data will not be completely overwritten by the new entry. If the data is not completely overwritten, it appears at the end of the new entry and is transmitted with the new entry to the host.

Rule

You must enable block mode to use clear mode.

Parameter Values

Refer to the following table on how to set p3 of the command string to disable or enable clear mode:

When you enter ...	Then clear mode is ...
0	off.
1 (default)	on.

Clear Mode Off

When clear mode is off, the client does not clear the data in a field. The new data entered overwrites portions of the left-over data. An operation that has bar code entries with equal character lengths might choose clear mode off.

Clear Mode On

When clear mode is on, the client clears the data entered in a field each time a new bar code entry occurs. An operation that has bar code entries with varying lengths would need to use clear mode on to ensure accurate transmission of each entry.

Truncate Mode: p4

Truncate mode determines whether the character positions of a bar code entry that exceeds the field length are simply dropped off or if the leftover characters are entered in the next available field.

Rule

You must enable block mode to use truncate mode.

Parameter Values

Refer to the following table on how to set p4 of the command string to disable or enable truncate mode:

When you enter ...	Then truncate mode is ...
0	off.
1 (default)	on.

Truncate Mode Off

When truncate mode is off, the character positions of a bar code entry exceeding the field length are entered in the next available field. An operation that has two kinds of information in one bar code entry would use truncate mode off so that one bar code read can fill two fields on a form.

Truncate Mode On

When truncate mode is on, the character positions of a bar code entry exceeding the field length are dropped off. If the field has 10 available character positions and the bar code entry has 22, the first 10 character positions of the entry are entered in the field and the remaining 12 character positions are dropped. An operation that wants to ensure that a bar code entry from one field is not carried over into the next field would want to have truncate mode on.

Advance Mode: p5

Advance mode determines whether the cursor advances to the next field at the end of a bar code entry. If the cursor does not advance to the next field, the next bar code entry is entered at the end of the previous bar code entry.

Rule

You must enable block mode to use advance mode.

Parameter Values

Refer to the following table on how to set p5 of the command string to disable or enable advance mode:

When you enter ...	Then advance mode is ...
0	off.
1 (default)	on.

Advance Mode Off

When advance mode is off, the cursor does not advance to the next field at the end of a bar code entry. The next bar code entry is entered directly following the previous bar code entry.

Advance Mode On

When advance mode is on, the cursor advances to the next field at the end of each entry regardless of whether the field is completely filled or not. An operation that has bar code entries that do not always fill the field would use advance mode on.

If autosend (LXEAS) is also enabled, the client will automatically send the screen even if the last bar code entry did not completely fill the last field on the display.

Specifying Transmit and Screen Function Setup

Command String for Transmit and Screen Function Setup

You must specify the transmit and screen function setup with the appropriate command string. This section discusses elements of the command string and its structure.

Command String

Use the following command string for the transmit and screen function setup:

```
CSIc1;p1;p2;p3;p4;p5;p6;p7;p8;p9;p10+F
```

Rule

Each parameter must be preceded by a semicolon (;).

You can use the semicolon as a placeholder if you want to change only one parameter. For example, you can send the following string to change only p5 of the command string:

```
CSI1;;;;;1+F
```

String Elements

The following table describes the different elements of the command string for the transmit and screen function setup:

String Element	Description	Related Sections
CSI	ANSI control sequence introducer (Hex 9B for 8-bit code or hex 1B5B for 7-bit code).	-
c1	A compatibility byte that ensures the terminal emulation understands the command string sent to it. The c1 is a revision level character for compatibility purposes, which increments with each revision. The original value for this parameter is 1.	-
;	A required delimiter that must precede all parameter entries.	-
P1	Not used.	
P2	8-Bit Code control from the host. *Note: The client can always accept either 7-bit or 8-bit codes from the host. The value of this parameter is therefore ignored.*	8-Bit Code Table

String Element	Description	Related Sections
P3	Auto Transmit Mode	LXEAS
	0 = Auto Transmit Mode is OFF. The [Enter] key must be pressed to transmit the data.	
	1 = Auto Transmit Mode is ON. The data is automatically transmitted when the last field is filled with a bar code entry, an RS-232 port entry or keyboard entry.	
P4	LF Transmission on Send	LNM
	0 = The [Enter] key places a carriage return in the data stream.	
	1 = The [Enter] key places a carriage return and line feed in the data stream.	
P5	Screen Mode	LXELM
	0 = Screen Mode is Block Mode.	
	1 = Screen Mode is Scroll Mode.	
P6	New Line Echo Mode	LXENE
	0 = New Line Echo Mode OFF.	
	1 = New Line Echo Mode ON.	
P7	Auto Wrap Mode	DECAWM
	0 = Auto Wrap Mode OFF.	
	1 = Auto Wrap Mode ON.	
P8	On Line Mode	-
	0 = Operate in On Line Mode. The [Enter] key is enabled to send messages to the host through the RF link.	
	1 = Operate in Local Mode. The [Enter] key is disabled.	
P9	Not used.	
p10	Transmit Mode	SRM
	0 = Transmit in Block or Scroll Mode.	
	1 = Transmit in Character Mode.	
+F	The final character of the ANSI command string. This combination indicates the end of the command string and identifies the command string as a transmit and screen function mode.	-

Local Edit Commands

Local edit commands enable a programmer to control data entry from the client. This section discusses the two types of local edit commands and the command string elements for each.

Command String for Local Edit Match Field

A local edit match field restricts user data entry to exact match values or ranges of accepted values. A programmer can use this feature to prevent incorrect part number or bin location entries.

Definitions

An *exact match field* is a field with predetermined values that must be matched exactly by user entry.

Local edit match field is a feature provided by LXE that enables the client to perform data comparisons between the match buffer in the client and the data entered by the client user. In other words, the client locally edits information as the client user enters data instead of the host performing the edit.

A *match buffer* is an area of memory used by the local edit match field command to restrict data entry by the client user.

Rules

Refer to the following rules for using local edit match field:

- You must specify a local edit field by placing the cursor at the beginning of the appropriate field and sending the correct command string.

- You must place match data between two identical LXE local edit command strings.

- You must enable Local echo. Refer to RM (parameter value 12) in this manual for more information on local echo.

- You must enable block mode. If character mode or scroll mode is enabled, local edit match fields are ignored.

- Use characters entered into the match buffer field that do not match the local edit wildcard symbols (defaults: ?, $, #, +, *) as exact match characters.

Command String

Use the following command string for specifying a local edit match field:

    CSId+A

String Elements

The following table describes the different elements of the command string for specifying a local edit match field:

String Element	Description
CSI	ANSI control sequence introducer (Hex 9B for 8-bit code or Hex 1B5B for 7-bit code).
d	Keyboard Input

0 (default) =	Display the keyboard input if user entry matches the requirements of the match buffer.
1 =	Do not display the keyboard input if user entry matches the requirements of the match buffer. Host input on these cells is displayed.

+A	The final character of the ANSI command sequence. This combination indicates the end of the command string and identifies the command string as a local edit match field.

Example

Refer to the following example of a local edit match field command:

```
printf(" _ _ _ _ _ _ ");     /*6 character fields*/
printf("\033[6D");            /*CUB-move cursor back 6*/
printf("\033[0+A");           /*turn on local edit setup mode*/
printf("P?$$$$");             /*part number format*/
printf("\033[0+A");           /*turn off local edit setup mode*/
printf("\033[6D");            /*CUB-move cursor back 6*/
gets(s)                       /*wait for user to enter data*/
```

Command String for Local Edit Wildcard

Local edit wildcard enables the programmer to change the default match symbols. Wildcard match values represent a range of acceptable user input. The host may change the definition of a particular wildcard at any time without affecting any previously set local edit match fields. (The TE stores the *type* of wildcard rather than the actual wildcard value.)

Command String

Use the following command string for local edit wildcard:

    CSIc1;p1;p2;p3;p4;p5;p6+G

Rule

The values you enter for the symbols must be in their decimal equivalent forms. Refer to the *8-Bit Code table* in this manual for these decimal values.

You can use the semicolon as a placeholder if you want to change only one parameter. For example, you can send the following string to change only p5 of the command string:

    CSI1;;;;;1+G

String Elements

The following table describes the different elements of the command string for specifying a local edit wildcard:

String Element	Description
CSI	ANSI control sequence introducer (Hex 9B for 8-bit code or Hex 1B5B for 7-bit code).
c1	A compatibility byte that ensures the terminal emulation understands the command string sent to it. The c1 is a revision level character for compatibility purposes, which will increment with each revision. The original value for this parameter is 1.
;	A required delimiter that must precede all parameter entries.
p1	Not used.
p2	? (Alpha Only Match) Alpha only match includes the following characters: A-Z, a-z, comma, period, dash, space.
p3	$ (Digits Only Match) Digits only match includes the following characters: 0-9.
p4	# (Numeric Only Match) Numeric only match includes the following characters: 0-9, plus, comma, period, dash, space.

String Element	Description
p5	+ (Alphanumeric Only Match) Alphanumeric only match includes the following characters: A-Z, a-z, 0-9, plus, comma, period, dash, space.
p6	* (Match All) Match all includes any GL or GR characters.
+G	The final characters of the ANSI command sequence. This combination indicates the end of the command string and identifies the command string as a local edit wildcard command.

Using Local Edit

Using the local edit command enables the programmer to control data entry at the client. The local edit field length can be as long as the virtual screen.

The client edits the field locally as data is entered and not when the field is exited. During the editing of a field, if the data fails the match test, the cursor does not move.

EXCEPTION *When a bar code entry fails the match test, the cursor goes to the first position where the character did not match the test. The remaining characters in the input field are checked against the match test.*

Programming Local Edit Fields

1. Position the cursor to the beginning of the local edit field.

2. Send the CSId+A command to put the TE in local edit setup mode.

3. Send the local edit match characters which could be either exact match or wildcards. These are sent as regular GL or GR characters. They will be stored internally in the TE and not displayed on the screen. The cursor position will automatically be updated after each char is received.

   Note: *Setting a screen cell as a local edit cell overrides any previous DAQ/SPA/SSA settings for that cell and vice versa.*

4. If desired, position the cursor to a new location and send more local edit match characters to define more fields. Repeat as necessary.

5. Send a matching CSId+A command to put the TE in normal mode.

6. If automatic tabbing is desired to the start of each field, the host must send the standard area or tab commands at the start of each local edit field. This can be accomplished even when the TE is in local edit mode.

7. The TE user may now enter data through keyboard or RS-232 inputs.

If the user input passes the local edit checks, the data will be displayed (if the local edit field was set up as CSI0+A). The data will not be displayed if the local edit field was set up as CSI1+A. In either case, the cursor position will advance by 1.

If the data does not pass the local edit checks, the TE will beep to inform the user and the cursor will not advance.

Requirements

The following requirements must be met for local edit to work properly:

- The client must be in block mode (LXELM).
- Local echo must be enabled in send-receive mode (SRM).
- The client must be an LXE mobile client to have local edit capabilities.

Chapter 7 5250

5250 LXE Commands

Input Device ID (*K)

The input device ID feature provides the host application developer with the ability to determine which LXE mobile client user input device and which barcode symbology was utilized to enter data in any field on the screen. When the 5250 host application is programmed to request this data from the mobile client, and when the input device ID feature is enabled at the client, the client sends this information to the 5250 host emulation. This feature can be used with the match field edit and scan and increment features, in addition to normal user input.

This feature is useful to monitor the use of automatic input devices to enforce the practice of not using the keyboard or keypad as an input device.

Description

There are two major components to this feature:

- The **input ID** field identifies the field that follows as an operator input field.
- The **operator input field** is the area on the screen where information is returned to the host application.

Input ID Field

The input ID field:

- must be preceded by a field attribute (FA) that designates the input ID field as unprotected (operator input capable).
- is designated with an * (asterisk) as the first character and a K as the second character.
- can have any display attribute including non-display.
- does not have to be on the same line as the operator input field, but it must be the first input field to precede it.
- cannot have any input fields defined on the screen map between it and its associated operator input field.
- must have the Modified Data Tag (MDT) set by the host application so it will return to the host when [Enter] is pressed.
- behaves exactly the same as any protected field on the LXE client device.

Terminal Operation

When the input ID feature is enabled in client setup, the RFTerm 5250 terminal emulation modifies the input ID field to behave as if it has the Bypass (protected) field attribute. The user can place the cursor in the input ID field with the arrow keys, but input is not allowed. If the user attempts input in this field, an error message will appear indicating that the field is protected. If the input ID feature is not enabled at the client device, RFTerm will treat input ID fields as normal input fields.

Examples

Example 1

The example illustrates the input ID feature used with a normal input field. In all cases, the field attribute (FA) preceding the input ID field (*K) designates an unprotected (input capable) field:
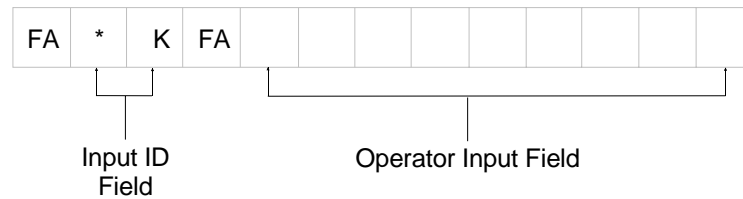
FA	*	K	FA									

Input ID
Field

Operator Input Field

Figure 7-1 Input ID Feature with Normal Input Field

After keyboard entry (123456789), the contents of the *input ID field* and the *operator input field* become:

FA	0	0	FA	1	2	3	4	5	6	7	8	9

Input ID
Field

Operator Input Field
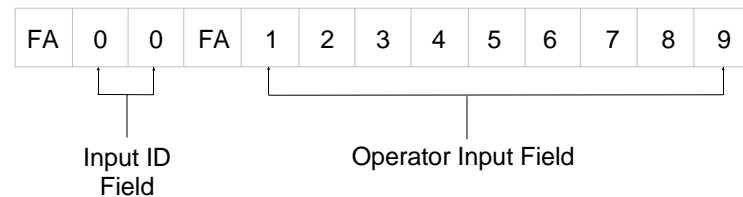
Figure 7-2 Contents of Fields after Keyboard Entry

The input ID field is now 00, which indicates that the user utilized the keyboard or keypad to enter data. In the event the user utilized a barcode scanner, the input ID field would contain a two-digit code identifying the barcode symbology. (Refer to the list of input device types and their corresponding input ID codes at the end of this segment.)

Input device ID reports the last device utilized by the end-user to fill in the input field to the host. If the user were to partially fill in an input field using a bar code scanner, and then finish filling in the input field with entry from the keyboard, the host application would report that entry for the field came from the keyboard or keypad. The user cannot modify the contents of the input ID field.

Example 2

Example 2 illustrates that protected fields can be present between the *input ID field* and the *operator input field*. The field attribute (FA) preceding ITEM: designates the prompt field as a protected field; the FA preceding the operator input field designates the last field as unprotected:
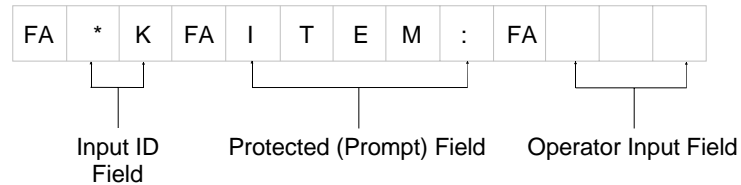
| FA | * | K | FA | I | T | E | M | : | FA | | |

Input ID   Protected (Prompt) Field   Operator Input Field
Field

Figure 7-3 Protected Fields

Input Identification Types

Here is a list of the numerical values that appear in the input ID field returned from the mobile client, designating the type of input device utilized by the end-user.

Numerical Value	Input Device Type
00	Keyboard / Keypad
01	Barcode Code 39
02	Barcode Code 128
03	Barcode Interleaved 2 of 5
04	Barcode UPC
05	Barcode EAN
06	Barcode Code 11
07	Barcode Discrete 2 of 5
08	Barcode MSI Plessey
50	RS-232

Figure 7-4 Input ID Types

Terminal Setup

You can configure printing options by sending the Terminal Setup command (*E) from the host to the client. These options apply when either the "Prnt" ToolBar button (IBM 3 or IBM 4 KeyBar) or the "Prt" button on the CE SmartPad is pressed. The Terminal Setup command has the following format, each parameter must be preceded by a semicolon (;) :

> *Ecl;p1;p2;p3;p4;p5;p6;p7;p8;p9;p10;p11;p12

This command must be sent to the client in the screen definition of the first field on the screen.

c1	A revision compatibility byte, currently not supported, use 1.
p1	Save to disk, currently not supported, use 0.
p2	Bar code left justify, currently not supported, use a semicolon as a place holder.
p3	Bar code input clear mode, currently not supported, use a semicolon as a place holder.
p4	Bar code input truncate mode, currently not supported, use a semicolon as a place holder.
p5	Bar code input truncate mode, currently not supported, use a semicolon as a place holder.

p6   Auto transmit after print.

> 0   Do not transmit an Enter command after a print (default)
> 1   Automatically transmit an Enter command after a print

p7   Print carriage return per line.

> 0   No carriage return per line on a screen print
> 1   Append a carriage return per line on a screen print (default)

p8   Print line feed per line.

> 0   No line feed per line on a screen print
> 1   Append a line feed per line on a screen print (default)

p9	Status line display mode, currently not supported, use a semicolon as a place holder.
p10	Input device ID, currently not supported, use a semicolon as a place holder.
p11	Swap Tab/Field Exit, currently not supported, use a semicolon as a place holder.
p12	Tab/Field Exit for advancing,, currently not supported, use a semicolon as a place holder.

Printing a Field (3270 and 5250)

You can print a field by sending a print command (*P) from the host to the client. It must be preceded by a field attribute (FA) byte. The printing ends when the next FA is encountered. The value of the FA is ignored, but it must be present.

Custom Mapping

The method for creating EBCDIC character sets for countries other than the U.S. is described below, and is followed by an example.

Custom Character Mapping

Upon loading, RFTerm first loads the standard character set **ISO Character Set 8859-1 (Latin 1)** found in the appendix. RFTerm then searches for a plain text document file named EBCDIC.TXT in the *\os dependent*\LXE\RFTerm folder, which is the default install folder for the application. If the file is found, RFTerm will read replacement character pairs from the EBCDIC.TXT file and place them in the substitution table for IBM emulations.

For example:

0x40 0x52

will replace the default character whose value is 0x40 with the character whose value is 0x52 according to ISO Character Set 8859-1 (Latin 1). Each occurrence of the character 0x40 will be replaced regardless of whether it originated from the host or from the client.

Where the `<white space>` represents any number of spaces and/or tabs, and is used for clarity. The first entry is the EBCDIC character position in hexadecimal in the *default EBCDIC character table*, and the second entry is the hexadecimal or decimal position in the ISO Character Set 8859-1 (Latin 1) table.

Custom Keyboard Mapping

RFTerm will also search for a KEYS.TXT file in the same folder. It will contain character pairs of the same format as the EBCDIC.TXT file, however these substitutions will be made **ONLY** for characters originating from the keyboard.

Note: If both files are present, a key press will be modified by the KEYS.TXT substitution and then modified by the EBCDIC.TXT substitution. Place a copy of the EBCDIC.TXT and KEYS.TXT files in the device operating system persistent folder. This will ensure the files are saved instead of discarded if the device is rebooted.

Appendix A Key Maps

The bitmaps and associated action text for all the actions and operations that can be associated with the configurable keys are contained in this appendix.

RFTerm ToolBar

Figure A-1 RFTerm ToolBar and KeyBar

←	Previous KeyBar		Configure Session	
→	Next KeyBar	Prt Scr	Print Screen	
	(Blank) no action		Start Menu Toggle	
Ent	Enter key		SmartPad Toggle	
Tab	Tab key		Text InputTool Toggle	
Btb	Field Backspace (Backtab)	×	Program Exit (Emulator)	
←	Left arrow key	F1	F1 function key	
→	Right arrow key	F2	F2 function key	
↑	Up arrow key	F3	F3 function key	
↓	Down arrow key	F4	F4 function key	
S1	Jump to Session 1 (S1)	F5	F5 function key	
S2	Jump to Session 2 (S2)	F6	F6 function key	
S3	Jump to Session 3 (S3)	F7	F7 function key	
S4	Jump to Session 4 (S4)	F8	F8 function key	
Ses	Jump to Next Connected Session	F9	F9 function key	
	Connect Session	F10	F10 function key	
A	Font Increase	F11	F11 function key	
A↓	Font Decrease	F12	F12 function key	

F13	F13 function key		F40	F40 function key (VT only)
F14	F14 function key		Atn	IBM Attention
F15	F15 function key		Bsp	IBM Backspace
F16	F16 function key		Clr	IBM Clear
F17	F17 function key		Del	IBM Delete
F18	F18 function key		Dup	IBM Duplicate
F19	F19 function key		Er EOF	IBM Erase EOF
F20	F20 function key		Er INP	IBM Erase Input
F21	F21 function key		Esc	IBM Escape key
F22	F22 function key		Fld +	IBM Field +
F23	F23 function key		Fld –	IBM Field –
F24	F24 function key		Fld EXIT	IBM Field Exit
F25	F25 function key (VT only)		Mrk	IBM Field Mark
F26	F26 function key (VT only)		Hlp	IBM Help (5250 only)
F27	F27 function key (VT only)		Hm	IBM Home
F28	F28 function key (VT only)		Ins	IBM Insert Toggle
F29	F29 function key (VT only)		NL	IBM New Line
F30	F30 function key (VT only)		OIA	IBM OIA Toggle
F31	F31 function key (VT only)		Prt	IBM Print (5250 only)
F32	F32 function key (VT only)		PA1	IBM PA1 key (3270 only)
F33	F33 function key (VT only)		PA2	IBM PA2 key (3270 only)
F34	F34 function key (VT only)		PA3	IBM PA3 key (3270 only)
F35	F35 function key (VT only)		Rst	IBM Reset
F36	F36 function key (VT only)		Roll ↑	IBM Roll Up (5250 only)
F37	F37 function key (VT only)		Roll ↓	IBM Roll Down (5250 only)
F38	F38 function key (VT only)		SRq	IBM System Request
F39	F39 function key (VT only)		Ans	VT Answerback

Bsp	VT Backspace		K9	VT Numpad 9
Can	VT Cancel		K,	VT Numpad Comma
Del	VT Delete		KE	VT Numpad Enter
Do	VT Do		K–	VT Numpad Minus
Esc	VT Escape		K.	VT Numpad Period
Fnd	VT Find		0	ASCII 0
Hlp	VT Help		1	ASCII 1
Hld	VT Hold		2	ASCII 2
Ins	VT Insert Here		3	ASCII 3
LF	VT Linefeed		4	ASCII 4
Nxt	VT Next		5	ASCII 5
PF1	VT PF1		6	ASCII 6
PF2	VT PF2		7	ASCII 7
PF3	VT PF3		8	ASCII 8
PF4	VT PF4		9	ASCII 9
Prv	VT Previous		+	ASCII +
Rem	VT Remove		–	ASCII -
Sel	VT Select		U6	VT UDK 6
K0	VT Numpad 0		U7	VT UDK 7
K1	VT Numpad 1		U8	VT UDK 8
K2	VT Numpad 2		U9	VT UDK 9
K3	VT Numpad 3		U10	VT UDK 10
K4	VT Numpad 4		U11	VT UDK 11
K5	VT Numpad 5		U12	VT UDK 12
K6	VT Numpad 6		U13	VT UDK 13
K7	VT Numpad 7		U14	VT UDK 14
K8	VT Numpad 8		U15	VT UDK 15

ᵘ16	VT UDK 16		ᴵ17	Input 17 (Text InputTool)
ᵘ17	VT UDK 17		ᴵ18	Input 18 (Text InputTool)
ᵘ18	VT UDK 18		ᴵ19	Input 19 (Text InputTool)
ᵘ19	VT UDK 19		ᴵ20	Input 20 (Text InputTool)
ᵘ20	VT UDK 20			Horizontal Scroll Toggle
ᴵ1	Input 1 (Text InputTool)			Vertical Scroll Toggle
ᴵ2	Input 2 (Text InputTool)			Horizontal Scroll Left (Page)
ᴵ3	Input 3 (Text InputTool)			Horizontal Scroll Right (Page)
ᴵ4	Input 4 (Text InputTool)			Vertical Scroll Up (Page)
ᴵ5	Input 5 (Text InputTool)			Vertical Scroll Down (Page)
ᴵ6	Input 6 (Text InputTool)			Horizontal Scroll Left (End)
ᴵ7	Input 7 (Text InputTool)			Horizontal Scroll Right (End)
ᴵ8	Input 8 (Text InputTool)			Vertical Scroll Up (End)
ᴵ9	Input 9 (Text InputTool)			Vertical Scroll Down (End)
ᴵ10	Input 10 (Text InputTool)			Scroll Upper Left quadrant
ᴵ11	Input 11 (Text InputTool)			Scroll Upper Right quadrant
ᴵ12	Input 12 (Text InputTool)			Scroll Lower Left quadrant
ᴵ13	Input 13 (Text InputTool)			Scroll Lower Right quadrant
ᴵ14	Input 14 (Text InputTool)			Scroll Center quadrant
ᴵ15	Input 15 (Text InputTool)			Scroll Cursor Center
ᴵ16	Input 16 (Text InputTool)			Scroll Cursor Visible

Figure A-2 RFTerm Buttons

RFTerm Functions

Function	Path
Connect to Host	Session | Connect or ToolBar Button
Disconnect from Host	Session | Disconnect or ToolBar Button
Hide / Display MenuBar	File | Settings | Display | Hide Main Menu
Enable / Disable MenuBar (XP)	File | Settings | Display | Disable MenuBar
Hide / Display ToolBar	File | Settings | Display | Hide ToolBar
Hide / Display KeyBar	File | Settings | Display | Hide KeyBar
Hide / Display All Bars**	File | Settings | Display | Hide All Bars
Toggle MenuBar Bottom / Top**	File | Settings | Display | Menu at Bottom
Configure Host Session	Session | Configure
Increase Font Size	ToolBar Button: Capital A with Up Arrow
Decrease Font Size	ToolBar Button: Capital A with Down Arrow
Font Selection	Session | Configure | Font | Font Weight combo box
Display / Hide SmartPad**	Display | SmartPad
Display / Hide InputTool**	Display | InputTool
Scroll Up	Scroll Bar
Scroll Down	Scroll Bar
Scroll Left	Scroll Bar
Scroll Right	Scroll Bar
Print Setup	File | Print Setup
Exit RFTerm	File | Exit

** Not available on Windows XP/2000 devices.

RFTerm Functions Not Supported on XP / 2000 Devices

Display / Hide SmartPad
Display / Hide InputTool
Toggle MenuBar Bottom / Top
Hide / Display All Bars

Virtual Keymaps

The **Virtual Key** tables indicate the key combinations that RFTerm looks for to perform each function. These key combinations are provided for the convenience of users who wish to remap RFTerm functions to physical keys different from those provided by the default key map. In most cases, the key combinations used by RFTerm are the same for each device type. Exceptions are noted.

RFTerm Functions – Virtual Keys

Important note for HX2 users:

When running an HX2 with an Alpha Mode 3 Tap keypad, RFTerm highjacks virtual function keys F11 through F19 to perform the RFTerm functions indicated in the "HX2" column of this table. This highjacking occurs only if the "HX2 Function Key Enable" option is not selected. If "HX2 Function Key Enable" is selected, the keymap reverts to the standard RFTerm virtual key map (indicated in the column labeled "Virtual Key" in the table below). Please see *HX2 Function Key Enable* in Chapter 4, *Session Configuration*, for details.

RFTerm Function	Virtual Key	HX2	RFTerm GUI Equivalent
Connect to Host	Ctrl Shift C	F17	Session | Connect
Disconnect from Host	Ctrl Shift D	F17	Session | Disconnect
Exit the Emulator	Alt X *or* Ctrl Shift X	Escape	File | Exit
Help	Ctrl H	Ctrl H	N/A
Hide / Display All Bars	Ctrl Shift L	F16	File | Settings | Display | Hide All Bars
Hide / Display MenuBar	Crtl Shift M	F15	File | Settings | Display | Hide MenuBar
Hide / Display ToolBar	Ctrl Shift T	F14	File | Settings | Display | Hide ToolBar
Hide / Display KeyBar	Ctrl Shift K	F13	File | Settings | Display | Hide KeyBar
Toggle MenuBar Bottom / Top	Ctrl Shift B	Ctrl Shift B	File | Settings | Display | Menu at Bottom
Configure Host Session	Ctrl Shift N	F18	Session | Configure
Application Settings	Ctrl Shift E	F19	File | Settings
Increase Font Size	Ctrl Shift Up Arrow	F12	Display | Font Up
Decrease Font Size	Ctrl Shift Down Arrow	F11	Display | Font Down
Toggle Font Bold / Normal	Ctrl Shift 1	Ctrl Shift 1	Session | Configure | Font tab
Display / Hide SmartPad	Ctrl Shift P	Ctrl Shift P	Display | SmartPad
Display / Hide InputTool	Ctrl Shift I	Ctrl Shift I	Display | InputTool
Scroll Up One Row	Shift Up Arrow	Shift Up Arrow	Scroll Bar
Scroll Down One Row	Shift Down Arrow	Shift Down Arrow	Scroll Bar

RFTerm Function	Virtual Key	HX2	RFTerm GUI Equivalent
Scroll Left One Row	Shift Left Arrow	Shift Left Arrow	Scroll Bar
Scroll Right One Row	Shift Right Arrow	Shift Right Arrow	Scroll Bar

ANSI Functions – Virtual Keys

When running the RFTerm ANSI on the HX2, please refer to the **ANSI Plus Reference Guide** for descriptions/explanations of the ANSI functions:

ANSI Function	Virtual Key	RFTerm GUI Equivalent
Answerback	Not Supported	Display	SmartPad	Ans
Backspace	BKSP	Display	SmartPad	BS
Delete	CTRL D	Display	SmartPad	Del
Exit Program	Alt X *or* CTRL Shift X	File	Exit
Function Key Editor	Not Supported	Session	Configure	FNKeys
Help	Ctrl H	
Local Echo On / Off	Not Supported	Session	Configure	Connection	VT Config	Screen	Local Echo
Print Screen	Ctrl Alt P	
Send	Enter	Display	Smartpad	Ent
Window Down	Ctrl Down Arrow	Scrollbar
Window Left	Ctrl Left Arrow	Scrollbar
Window Right	Ctrl Right Arrow	Scrollbar
Window Up	Ctrl Up Arrow	Scrollbar
F1 – F10	F1 – F10	Display	SmartPad	F1 – F10
F11 – F20	F11 – F20 [2]	Display	SmartPad	F11 – F20 [2]
F21 – F30	F21 – F30	Display	SmartPad	F21 – F30
F21 – F40	F31 – F40	Display	SmartPad	F31 – F40

---

[2] On the HX2 with original keypad, virtual keys F11 – F19 are used to implement RFTerm functions. This behavior renders F11 – F19 functions inaccessible to RFTerm. However, when "HX2 Function Key Enable" is selected, the RFTerm internal keymapping is changed so the RFTerm functions are no longer assigned to the F11 – F19 virtual keys. Please see *HX2 Function Key Enable* in Chapter 4, *Session Configuration*, for details.

IBM 3270 Functions – Virtual Keys

When running the RFTerm IBM 3270 terminal emulator on the HX2, please refer to the **TN3270 Terminal Emulation Reference Guide** for descriptions/explanations of the 3270 functions.

3270 Function	Virtual Key	RFTerm GUI Equivalent
Attention	Ctrl A	Display	SmartPad	Attn
Backtab	Shift Tab	Display	SmartPad	Btb
Clear	Ctrl C	Display	SmartPad	Clr
Cursor Block / Underline	Not Supported	File	Settings	Cursor	Cursor Options
Delete	Ctrl D	Display	SmartPad	Del
Erase	Bksp	Display	SmartPad	BSp
Erase Input	Ctrl Bksp	Display	SmartPad	Ein
Error Reset	Ctrl R	Display	SmartPad	Rst
Exit Program	Alt X *or* Ctrl Shift X	File	Exit
Function Key Editor	Not Supported	Session	Configure	FNKeys
Help	Ctrl H	
Home Key	Not Supported	Display	SmartPad	Hm
Insert	Ctrl I	Display	SmartPad	Ins
New Line Key	Ctrl N	Display	SmartPad	NL
Next or Tab	Tab	Display	SmartPad	Tb
Print Screen	Ctrl Alt P	
Roll Up	Not Supported	IBM 3 KeyBar: Roll Up
Roll Down	Not Supported	IBM 3 KeyBar: Roll Dn
SW Rev	Not Supported	File	About
Send	Not Supported	Base KeyBar: Ent
Status Line Toggle	Alt T *or* Ctrl Shift O	File	Settings	Display	Hide OIA
System Request	Ctrl S	Display	SmartPad	Srq
Window Down	Ctrl Down Arrow	Scrollbar
Window Right	Ctrl Right Arrow	Scrollbar
Window Left	Ctrl Left Arrow	Scrollbar
Window Up	Ctrl Up Arrow	Scrollbar
PA1 – PA3	Ctrl 1 – 3	Display	SmartPad	Pa1 – Pa3
F1 – F10	F1 – F10	Display	SmartPad	F1 – F10

3270 Function	Virtual Key	RFTerm GUI Equivalent
F11 – F20	F11 – F20 [3]	Display | SmartPad | F1 – F20 [3]
F21 – F24	F21 – F24	Display | SmartPad | F21 – F24

IBM 5250 Functions – Virtual Keys

When running the RFTerm IBM 5250 terminal emulator on HX2 devices, please refer to the **TN5250 Terminal Emulation Reference Guide** for descriptions/explanations of the 5250 functions.

5250 Function	Virtual key	RFTerm GUI Equivalent
Attention	Ctrl A	Display | SmartPad | Attn
Backtab	Shift Tab	Base KeyBar: Btb
Char Backspace	Bksp	Display | SmartPad | BSp
Clear	Ctrl C	Display | SmartPad | Clr
Cursor Block / Underline	Not Supported	File | Settings | Cursor | Cursor Options
Delete	Ctrl D	Display | SmartPad | Del
Dup	Ctrl U	Display | SmartPad | Dup
Erase Input	Ctrl Bksp	Display | SmartPad | Ein
Error Reset or Reset	Esc	Display | SmartPad | Rst
Exit Program	Alt X *or* Ctrl Shift X	File | Exit
Field Exit	Ctrl Enter	Display | SmartPad | Fex
Field Minus	Ctrl M	Display | SmartPad | F-
Field Plus	Ctrl L	Display | SmartPad | F+
Function Key Editor	Not Supported	Session | Configure | FNKeys
Home Key	Not Supported	Display | SmartPad | Hm
Help	Ctrl H	
Insert	Ctrl I	Display | SmartPad | Ins
New Line Key	Ctrl N	Display | SmartPad | NL
Next or Field Advance or Tab	Tab	Base KeyBar: Tb
Print Screen	Ctrl Alt P	Display | SmartPad | Prt
Send or Enter CATV	Enter	Base KeyBar: Ent

---

[3] On the HX2 with original keypad, virtual keys F11 – F19 are used to implement RFTerm functions. This behavior renders F11 – F19 functions inaccessible to RFTerm. However, when "HX2 Function Key Enable" is selected, the RFTerm internal keymapping is changed so the RFTerm functions are no longer assigned to the F11 – F19 virtual keys. Please see *HX2 Function Key Enable* in Chapter 4, *Session Configuration*, for details.

5250 Function	Virtual key	RFTerm GUI Equivalent
Status Line Toggle	Alt T *or* Ctrl Shift 0	Special KeyBar: OIA
SW Rev	Not Supported	File | About
System Request	Ctrl S	Display | SmartPad | Srq
Window Down	Ctrl Down Arrow	Scrollbar
Window Left	Ctrl Left Arrow	Scrollbar
Window Right	Ctrl Right Arrow	Scrollbar
Window Up	Ctrl Up Arrow	Scrollbar
F1 – F10	F1 – F10	Display | SmartPad | F1 – F10
F11 – F20	F11 – F20 [4]	Display | SmartPad | F11 – F20 [4]
F21 – F24	F21 – F24	Display | SmartPad | F21 – F24

---

[4] On the HX2 with original keypad, virtual keys F11 – F19 are used to implement RFTerm functions. This behavior renders F11 – F19 functions inaccessible to RFTerm. However, when "HX2 Function Key Enable" is selected, the RFTerm internal keymapping is changed so the RFTerm functions are no longer assigned to the F11 – F19 virtual keys. Please see *HX2 Function Key Enable* in Chapter 4, *Session Configuration*, for details.

HX2 Key Maps

Alpha Mode 3 Tap Keypad

Dual Alpha Keypad

Triple Tap Keypad

Figure A-3 HX2 Keypads

Note: All RFTerm shortcut keys are directly accessible through the RFTerm user interface.

When running RFTerm on the HX2, please refer to the HX2 Reference Guide for key function information / descriptions.

When using a sequence of keys that includes the Blue key, press the color key first then the rest of the key sequence. The Blue key is a "sticky key". It does not need to be held down before pressing the next (or desired) key.

User defined keys can be set using **Start | Settings | Control Panel | Keypad** and selecting the **KeyMap** tab. Please see the HX2 Reference Guide for details.

Note: When the computer boots, the default condition of Caps (or CapsLock) is Off. The Caps (or CapsLock) condition can be toggled with Blue plus Tab key sequence.

RFTerm Functions – HX2

Alpha Mode 3 Tap Keypad – HX2

The notation "Not Defined" in the following table indicates that no key combination provides this function in the default key map. To obtain this function, some key (or combination of keys) must be remapped to provide key combination specified earlier in the "RFTerm Functions – Virtual Keys" table, earlier in this Appendix.

RFTerm Function	Press this key and then…	Press this HX2 key	RFTerm GUI Equivalent
Connect to Host	Blue	7	Session | Connect
Disconnect from Host	Blue	7	Session | Disconnect
Exit the Emulator	Blue	Bkspc	File | Exit
Help	Not Defined		N/A
Hide / Display All Bars	Blue	6	File | Settings | Display | Hide All Bars
Hide / Display MenuBar	Blue	5	File | Settings | Display | Hide MenuBar
Hide / Display ToolBar	Blue	4	File | Settings | Display | Hide ToolBar
Hide / Display KeyBar	Blue	3	File | Settings | Display | Hide KeyBar
Toggle MenuBar Bottom / Top	Not Defined		File | Settings | Display | Menu at Bottom
Configure Host Session	Blue	8	Session | Configure
Application Settings	Blue	9	File | Settings
Increase Font Size	Blue	2	Display | Font Up
Decrease Font Size	Blue	1	Display | Font Down
Toggle Font Bold / Normal	Not Defined		Session | Configure | Font tab
Display / Hide SmartPad	Not Defined		Display | SmartPad
Display / Hide InputTool	Not Defined		Display | InputTool
Scroll Up One Row	Not Defined		Scroll Bar
Scroll Down One Row	Not Defined		Scroll Bar
Scroll Left One Row	Not Defined		Scroll Bar
Scroll Right One Row	Not Defined		Scroll Bar

Dual Alpha Keypad – HX2

RFTerm Function	Press these HX2 keys in this order…				RFTerm GUI Equivalent
Connect to Host	Ctrl	Shift	Orange	2	Session | Connect
Disconnect from Host	Ctrl	Shift	Lt Blue	2	Session | Disconnect
Exit the Emulator	Green	Ctrl	Lt Blue	Bksp	File | Exit
Help	Ctrl	Lt Blue	4		N/A
Hide / Display All Bars	Ctrl	Shift	Lt Blue	2	File | Settings | Display | Hide All Bars
Hide / Display MenuBar	Ctrl	Shift	Orange	7	File | Settings | Display | Hide MenuBar
Hide / Display ToolBar	Ctrl	Shift	Lt Blue	Up Arrow	File | Settings | Display | Hide ToolBar
Hide / Display KeyBar	Ctrl	Shift	Orange	6	File | Settings | Display | Hide KeyBar
Toggle MenuBar Bottom / Top	Ctrl	Shift	Lt Blue	1	File | Settings | Display | Menu at Bottom
Configure Host Session	Ctrl	Shift	Lt Blue	7	Session | Configure
Application Settings	Ctrl	Shift	Orange	3	File | Settings
Increase Font Size	Ctrl	Shift	Up Arrow		Display | Font Up
Decrease Font Size	Ctrl	Shift	Down Arrow		Display | Font Down
Toggle Font Bold / Normal	Ctrl	Shift	1		Session | Configure | Font tab
Display / Hide SmartPad	Ctrl	Shift	Lt Blue	8	Display | SmartPad
Display / Hide InputTool	Ctrl	Shift	Orange	5	Display | InputTool
Scroll Up One Row	Shift	Up Arrow			Scroll Bar
Scroll Down One Row	Shift	Down Arrow			Scroll Bar
Scroll Left One Row	Shift	Green	Up Arrow		Scroll Bar
Scroll Right One Row	Shift	Green	Down Arrow		Scroll Bar

Triple Tap Keypad – HX2

RFTerm Function	Press these HX2 keys in this order…						RFTerm GUI Equivalent
Connect to Host	Ctrl	Shift	Lt Blue	2	2	2	Session | Connect
Disconnect from Host	Ctrl	Shift	Lt Blue	3			Session | Disconnect
Exit the Emulator	Green	Ctrl	Lt Blue	9	9		File | Exit
Help	Ctrl	Lt Blue	4	4			N/A
Hide / Display All Bars	Ctrl	Shift	Lt Blue	5	5	5	File | Settings | Display | Hide All Bars
Hide / Display MenuBar	Ctrl	Shift	Lt Blue	6			File | Settings | Display | Hide MenuBar
Hide / Display ToolBar	Ctrl	Shift	Lt Blue	8			File | Settings | Display | Hide ToolBar
Hide / Display KeyBar	Ctrl	Shift	Lt Blue	5	5		File | Settings | Display | Hide KeyBar
Toggle MenuBar Bottom / Top	Ctrl	Shift	Lt Blue	2	2		File | Settings | Display | Menu at Bottom
Configure Host Session	Ctrl	Shift	Lt Blue	6	6		Session | Configure
Application Settings	Ctrl	Shift	Lt Blue	3	3		File | Settings
Increase Font Size	Ctrl	Shift	Up Arrow				Display | Font Up
Decrease Font Size	Ctrl	Shift	Down Arrow				Display | Font Down
Toggle Font Bold / Normal	Ctrl	Shift 1					Session | Configure | Font tab
Display / Hide SmartPad	Ctrl	Shift	Lt Blue	7			Display | SmartPad
Display / Hide InputTool	Ctrl	Shift	Lt Blue	4	4	4	Display | InputTool
Scroll Up One Row	Shift	Up Arrow					Scroll Bar
Scroll Down One Row	Shift	Down Arrow					Scroll Bar
Scroll Left One Row	Shift	Green	Up Arrow				Scroll Bar
Scroll Right One Row	Shift	Green	Down Arrow				Scroll Bar

ANSI Functions – HX2

When running the RFTerm ANSI on the HX2, please refer to the **ANSI Plus Reference Guide** for descriptions/explanations of the ANSI functions.

Alpha Mode 3 Tap Keypad – HX2

The notation "Not Defined" in the following table indicates that no key combination provides this function in the default key map. To obtain this function, some key (or combination of keys) must be remapped to provide key combination specified earlier in the "ANSI Functions – Virtual Keys" table, earlier in this Appendix.

When running an HX2 with an Alpah Mode 3 Tap keypad, RFTerm highjacks virtual function keys F11 through F19 to perform the RFTerm functions indicated in the "RFTerm Functions – HX2" section for the Alpha Mode 3 Tap keypad. This highjacking occurs only if the "HX2 Function Key Enable" option is not selected. If "HX2 Function Key Enable" is selected, the keymap reverts to the standard RFTerm virtual key map (indicated in the column labeled "Virtual Key" in the table below). Please see *HX2 Function Key Enable* in Chapter 4, *Session Configuration*, for details.

ANSI Function	Press this HX2 key and then…	Press this HX2 key	RFTerm GUI Equivalent
Answerback	Not Supported		Display	SmartPad	Ans
Backspace		Bkspc	Display	SmartPad	BS
Delete	Not Defined		Display	SmartPad	Del
Exit Program	Blue	Bkspc	File	Exit
Function Key Editor	Not Supported		Session	Configure	FNKeys
Help	Not Defined		
Local Echo On / Off	Not Supported		Session	Configure	Connection	VT Config	Screen	Local Echo
Print Screen	Not Defined		
Send		Enter	Display	Smartpad	Ent
Window Down	Not Defined		Scrollbar
Window Left	Not Defined		Scrollbar
Window Right	Not Defined		Scrollbar
Window Up	Not Defined		Scrollbar
F1 – F4		F1 – F4	Display	SmartPad	F1 – F4
F5 – F8	Blue	F1 – F4	Display	SmartPad	F5 – F8
F9	Not Defined		Display	SmartPad	F9
F10	Blue	0	Display	SmartPad	F10
F11 – F19	Blue	1 – 9	Display	SmartPad	F11 – F19
F20	Not Defined		Display	SmartPad	F20
F21 – F40	Not Defined		KeyBar

Dual Alpha Keypad – HX2

ANSI Function	Press these HX2 keys in this order…				RFTerm GUI Equivalent
Answerback	Not supported				Display | SmartPad | Ans
Backspace	Bksp				Display | SmartPad | BS
Delete	Ctrl	Lt Blue	2		Display | SmartPad | Del
Exit Program	Green	Ctrl	Lt Blue	Bksp	File | Exit
Function Key Editor	Not supported				Session | Configure | FNKeys
Help	Ctrl	Lt Blue	4		
Local Echo On / Off	Not supported				Session | Configure | Connection | VT Config | Screen | Local Echo
Print Screen	Green	Ctrl	Lt Blue	8	
Send	Enter				Display | Smartpad | Ent
Window Down	Ctrl	Down Arrow			Scrollbar
Window Left	Ctrl	Green	Up Arrow		Scrollbar
Window Right	Ctrl	Green	Down Arrow		Scrollbar
Window Up	Ctrl	Up Arrow			Scrollbar
F1 – F9	Green	1 – 9			Display | SmartPad | F1 – F9
F10	Green	0			Display | SmartPad | F10
F11 – F19	Shift	Green	1 – 9		Display | SmartPad | F11 – F19
F20	Shift	Green	0		Display | SmartPad | F20
F21 – F24	Green	Orange	1 – 4		Display | SmartPad | F21 – F24
F25 – F40	Not supported				Display | SmartPad | F25 – F40

Triple Tap Keypad – HX2

ANSI Function	Press these HX2 keys in this order…					RFTerm GUI Equivalent
Answerback	Not supported					Display	SmartPad	Ans
Backspace	Bksp					Display	SmartPad	BS
Delete	Ctrl	Lt Blue	3			Display	SmartPad	Del
Exit Program	Green	Ctrl	Lt Blue	9	9	File	Exit
Function Key Editor	Not supported					Session	Configure	FNKeys
Help	Ctrl	Lt Blue	4	4		
Local Echo On / Off	Not supported					Session	Configure	Connection	VT Config	Screen	Local Echo
Print Screen	Green	Ctrl	Lt Blue	7		
Send	Enter					Display	Smartpad	Ent
Window Down	Ctrl	Down Arrow				Scrollbar
Window Left	Ctrl	Green	Up Arrow			Scrollbar
Window Right	Ctrl	Green	Down Arrow			Scrollbar
Window Up	Ctrl	Up Arrow				Scrollbar
F1 – F9	Green	1 – 9				Display	SmartPad	F1 – F9
F10	Green	0				Display	SmartPad	F10
F11 – F19	Shift	Green	1 – 9			Display	SmartPad	F11 – F19
F20	Shift	Green	0			Display	SmartPad	F20
F21 – F24	Shift	Green	Lt Blue	1 – 4		Display	SmartPad	F21 – F24
F25 – F40	Not supported					Display	SmartPad	F25 – F40

IBM 3270 Functions – HX2

When running the RFTerm IBM 3270 terminal emulator on the HX2, please refer to the **TN3270 Terminal Emulation Reference Guide** for descriptions/explanations of the 3270 functions.

Alpha Mode 3 Tap Keypad – HX2

The notation "Not Defined" in the following table indicates that no key combination provides this function in the default key map. To obtain this function, some key (or combination of keys) must be remapped to provide key combination specified earlier in the "IBM 3270 Functions – Virtual Keys" table, earlier in this Appendix.

When running an HX2 with an Alpha Mode 3 Tap keypad, RFTerm highjacks virtual function keys F11 through F19 to perform the RFTerm functions indicated in the "RFTerm Functions – HX2" section for the Alpha Mode 3 Tap keypad. This highjacking occurs only if the "HX2 Function Key Enable" option is not selected. If "HX2 Function Key Enable" is selected, the keymap reverts to the standard RFTerm virtual key map (indicated in the column labeled "Virtual Key" in the table below). Please see *HX2 Function Key Enable* in Chapter 4, *Session Configuration*, for details.

3270 Function	Press this HX2 key and then…	Press this HX2 key	RFTerm GUI Equivalent
Attention	Not Defined		Display | SmartPad | Attn
Backtab	Blue	Left Arrow	Display | SmartPad | Btb
Clear	Not Defined		Display | SmartPad | Clr
Cursor Block / Underline	Not Supported		File | Settings | Cursor | Cursor Options
Delete	Not Defined		Display | SmartPad | Del
Erase		Bkspc	Display | SmartPad | BSp
Erase Input	Not Defined		Display | SmartPad | Ein
Error Reset	Not Defined		Display | SmartPad | Rst
Exit Program	Blue	Bkspc	File | Exit
Function Key Editor	Not Supported		Session | Configure | FNKeys
Help	Not Defined		
Home Key	Not Defined		Display | SmartPad | Hm
Insert	Not Defined		Display | SmartPad | Ins
New Line Key	Not Defined		Display | SmartPad | NL
Next or Tab	Blue	Right Arrow	Display | SmartPad | Tb
Print Screen	Not Defined		
Roll Up	Not Supported		IBM 3 KeyBar: Roll Up
Roll Down	Not Supported		IBM 3 KeyBar: Roll Dn
SW Rev	Not Defined		File | About
Send	Not Supported		Base KeyBar: Ent
Status Line Toggle	Not Defined		File | Settings | Display | Hide OIA
System Request	Not Defined		Display | SmartPad | Srq
Window Down	Not Defined		Scrollbar

3270 Function	Press this HX2 key and then…	Press this HX2 key	RFTerm GUI Equivalent
Window Right	Not Defined		Scrollbar
Window Left	Not Defined		Scrollbar
Window Up	Not Defined		Scrollbar
PA1 – PA3	Not Defined		Display	SmartPad	Pa1 – Pa3
F1 – F4		F1 – F4	Display	SmartPad	F1 – F4
F5 – F8	Blue	F1 – F4	Display	SmartPad	F5 – F8
F9	Not Defined		Display	SmartPad	F9
F10	Blue	0	Display	SmartPad	F10
F11 – F19	Blue	1 – 9	Display	SmartPad	F11 – F19
F20 – F24	Not Defined		Display	SmartPad	F20 – F24

Dual Alpha Keypad– HX2

3270 Function	Press these HX2 keys in this order…				RFTerm GUI Equivalent
Attention	Ctrl	Orange	1		Display	SmartPad	Attn
Backtab	Green	Tab			Display	SmartPad	Btb
Clear	Ctrl	Orange	2		Display	SmartPad	Clr
Cursor Block / Underline	Not supported				File	Settings	Cursor	Cursor Options
Delete	Ctrl	Lt Blue	2		Display	SmartPad	Del
Erase	Bksp				Display	SmartPad	BSp
Erase Input	Ctrl Bksp				Display	SmartPad	Ein
Error Reset	Ctrl	Lt Blue	9		Display	SmartPad	Rst
Exit Program	Green	Ctrl	Lt Blue	Bksp	File	Exit
Function Key Editor	Not supported				Session	Configure	FNKeys
Help	Ctrl	Lt Blue	4		
Home Key	Not supported				Display	SmartPad	Hm
Insert	Ctrl	Orange	5		Display	SmartPad	Ins
New Line Key	Ctrl	Lt Blue	7		Display	SmartPad	NL
Next or Tab	Tab				Display	SmartPad	Tb
Print Screen	Green	Ctrl	Lt Blue	8	
Roll Up	Not supported				IBM 3 KeyBar: Roll Up
Roll Down	Not supported				IBM 3 KeyBar: Roll Dn
SW Rev	Not supported				File	About
Send	Not supported				Base KeyBar: Ent

3270 Function	Press these HX2 keys in this order…				RFTerm GUI Equivalent
Status Line Toggle	Green	Ctrl	Lt Blue	Up Arrow	File	Settings	Display	Hide OIA
System Request	Ctrl	Orange	Up Arrow		Display	SmartPad	Srq
Window Down	Ctrl	Down Arrow			Scrollbar
Window Right	Ctrl	Green	Right Arrow		Scrollbar
Window Left	Ctrl	Green	Left Arrow		Scrollbar
Window Up	Ctrl	Up Arrow			Scrollbar
PA1 – PA3	Ctrl	1			Display	SmartPad	Pa1 – Pa3
F1 – F9	Green	1 – 9			Display	SmartPad	F1 – F9
F10	Green	0			Display	SmartPad	F10
F11 – F19	Shift	Green	1 – 9		Display	SmartPad	F11 – F19
F20	Shift	Green	0		Display	SmartPad	F20
F20 – F24	Shift	Green	Lt Blue	1 – 4	Display	SmartPad	F21 – F24

Triple Tap Keypad – HX2

3270 Function	Press these HX2 keys in this order…					RFTerm GUI Equivalent
Attention	Ctrl	Lt Blue	2			Display	SmartPad	Attn
Backtab	Green	Tab				Display	SmartPad	Btb
Clear	Ctrl	Lt Blue	2	2	2	Display	SmartPad	Clr
Cursor Block / Underline						File	Settings	Cursor	Cursor Options
Delete	Ctrl	Lt Blue	3			Display	SmartPad	Del
Erase	Bksp					Display	SmartPad	BSp
Erase Input	Ctrl	Bksp				Display	SmartPad	Ein
Error Reset	Ctrl	Lt Blue	7	7	7	Display	SmartPad	Rst
Exit Program	Green	Ctrl	Lt Blue	9	9	File	Exit
Function Key Editor	Not supported					Session	Configure	FNKeys

3270 Function	Press these HX2 keys in this order…					RFTerm GUI Equivalent
Help	Ctrl	Lt Blue	4	4		
Home Key						Display | SmartPad | Hm
Insert	Ctrl	Lt Blue	4	4	4	Display | SmartPad | Ins
New Line Key	Ctrl	Lt Blue	6	6		Display | SmartPad | NL
Next or Tab	Tab					Display | SmartPad | Tb
Print Screen	Green	Ctrl	Lt Blue	8		
Roll Up	Not supported					IBM 3 KeyBar: Roll Up
Roll Down	Not supported					IBM 3 KeyBar: Roll Dn
SW Rev	Not supported					File | About
Send	Not supported					Base KeyBar: Ent
Status Line Toggle	Green	Ctrl	Lt Blue	8		File | Settings | Display | Hide OIA
System Request	Ctrl	Lt Blue	7	7	7 7	Display | SmartPad | Srq
Window Down	Ctrl	Down Arrow				Scrollbar
Window Right	Ctrl	Green	Down Arrow			Scrollbar
Window Left	Ctrl	Green	Up Arrow			Scrollbar
Window Up	Ctrl	Up Arrow				Scrollbar
PA1 – PA3	Ctrl	1 – 3				Display | SmartPad | Pa1 – Pa3
F1 – F9	Green	1 – 9				Display | SmartPad | F1 – F9
F10	Green	0				Display | SmartPad | F10
F11 – F19	Shift	Green	1 – 9			Display | SmartPad | F10 – F19
F20	Shift	Green	0			Display | SmartPad | F20
F21 – F24	Shift	Green	Lt Blue	1 – 4		Display | SmartPad | F21 – F24

IBM 5250 Functions – HX2

When running the RFTerm IBM 5250 terminal emulator on HX2 devices, please refer to the **TN5250 Terminal Emulation Reference Guide** for descriptions/explanations of the 5250 functions.

Alpha Mode 3 Tap Keypad – HX2

The notation "Not Defined" in the following table indicates that no key combination provides this function in the default key map. To obtain this function, some key (or combination of keys) must be remapped to provide key combination specified earlier in the "IBM 5250 Functions – Virtual Keys" table, earlier in this Appendix.

When running an HX2 with an Alpah Mode 3 Tap keypad, RFTerm highjacks virtual function keys F11 through F19 to perform the RFTerm functions indicated in the "RFTerm Functions – HX2" section for the Alpha Mode 3 Tap keypad. This highjacking occurs only if the "HX2 Function Key Enable" option is not selected. If "HX2 Function Key Enable" is selected, the keymap reverts to the standard RFTerm virtual key map (indicated in the column labeled "Virtual Key" in the table below). Please see *HX2 Function Key Enable* in Chapter 4, *Session Configuration*, for details.

5250 Function	Press this HX2 key and then…	Press this HX2 key	RFTerm GUI Equivalent
Attention	Not Defined		Display	SmartPad	Attn
Backtab	Blue	Left Arrow	Base KeyBar: Btb
Char Backspace		Bkspc	Display	SmartPad	BSp
Clear	Not Defined		Display	SmartPad	Clr
Cursor Block / Underline	Not Supported		File	Settings	Cursor	Cursor Options
Delete	Not Defined		Display	SmartPad	Del
Dup	Not Defined		Display	SmartPad	Dup
Erase Input	Not Defined		Display	SmartPad	Ein
Error Reset or Reset	Not Defined		Display	SmartPad	Rst
Exit Program	Blue	Bkspc	File	Exit
Field Exit	Blue	Enter	Display	SmartPad	Fex
Field Minus	Not Defined		Display	SmartPad	F-
Field Plus	Not Defined		Display	SmartPad	F+
Function Key Editor	Not Supported		Session	Configure	FNKeys
Home Key	Not Defined		Display	SmartPad	Hm
Help	Not Defined		
Insert	Not Defined		Display	SmartPad	Ins
New Line Key	Not Defined		Display	SmartPad	NL
Next or Field Advance or Tab	Blue	Right Arrow	Base KeyBar: Tb
Print Screen	Not Defined		Display	SmartPad	Prt
Send or Enter CATV		Enter	Base KeyBar: Ent
Status Line Toggle	Not Defined		Special KeyBar: OIA
SW Rev	Not Defined		File	About

5250 Function	Press this HX2 key and then…	Press this HX2 key	RFTerm GUI Equivalent
System Request	Not Defined		Display | SmartPad | Srq
Window Down	Not Defined		Scrollbar
Window Left	Not Defined		Scrollbar
Window Right	Not Defined		Scrollbar
Window Up	Not Defined		Scrollbar
F1 – F4		F1 – F4	Display | SmartPad | F1 – F4
F5 – F8	Blue	F1 – F4	Display | SmartPad | F5 – F8
F9	Not Defined		Display | SmartPad | F9
F10	Blue	0	Display | SmartPad | F10
F11 – F19	Blue	1 – 9	Display | SmartPad | F11 – F19
F20 – F24	Not Defined		Display | SmartPad | F20 – F24

Dual Alpha Keypad – HX2

5250 Function	Press these HX2 keys in this order…				RFTerm GUI Equivalent
Attention	Ctrl	Orange	1		Display | SmartPad | Attn
Backtab	Green	Tab			Base KeyBar: Btb
Char Backspace	Bksp				Display | SmartPad | BSp
Clear	Ctrl	Orange	Bksp		Display | SmartPad | Clr
Cursor Block / Underline	Not supported				File | Settings | Cursor | Cursor Options
Delete	Ctrl	Lt Blue	2		Display | SmartPad | Del
Dup	Ctrl	Orange	0		Display | SmartPad | Dup
Erase Input	Ctrl	Bksp			Display | SmartPad | Ein
Error Reset or Reset	Esc				Display | SmartPad | Rst
Exit Program	Green	Ctrl	Lt Blue	Bksp	File | Exit
Field Exit	Ctrl	Enter			Display | SmartPad | Fex
Field Minus	Ctrl	Orange	7		Display | SmartPad | F-
Field Plus	Ctrl	Lt Blue	6		Display | SmartPad | F+
Function Key Editor	Not supported				Session | Configure | FNKeys
Home Key	Not supported				Display | SmartPad | Hm
Help	Ctrl	Lt Blue	4		
Insert	Ctrl	Orange	5		Display | SmartPad | Ins
New Line Key	Ctrl	Lt Blue	7		Display | SmartPad | NL
Next or Field Advance or Tab	Tab				Base KeyBar: Tb

5250 Function	Press these HX2 keys in this order…				RFTerm GUI Equivalent
Print Screen	Green	Ctrl	Lt Blue	8	Display | SmartPad | Prt
Send or Enter CATV	Enter				Base KeyBar: Ent
Status Line Toggle	Green	Ctrl	Lt Blue	Up Arrow	Special KeyBar: OIA
SW Rev	Not supported				File | About
System Request	Ctrl	Orange	Up Arrow		Display | SmartPad | Srq
Window Down	Ctrl	Down Arrow			Scrollbar
Window Left	Ctrl	Green	Up Arrow		Scrollbar
Window Right	Ctrl	Green	Down Arrow		Scrollbar
Window Up	Ctrl	Up Arrow			Scrollbar
F1 – F9	Green	1 – 9			Display | SmartPad | F1 – F9
F10	Green	0			Display | SmartPad | F10
F11 – F19	Shift	Green	1 – 9		Display | SmartPad | F11 – F19
F20	Shift	Green	0		Display | SmartPad | F20
F21 – F24	Shift	Green	Lt Blue	1 – 4	Display | SmartPad | F21 – F24

Triple Tap Keypad – HX2

5250 Function	Press these HX2 keys in this order…				RFTerm GUI Equivalent
Attention	Ctrl	Lt Blue	2		Display | SmartPad | Attn
Backtab	Green	Tab			Base KeyBar: Btb
Char Backspace	Bksp				Display | SmartPad | BSp
Clear	Ctrl	Lt Blue	2	2	2	Display | SmartPad | Clr
Cursor Block / Underline	Not supported				File | Settings | Cursor | Cursor Options
Delete	Ctrl	Lt Blue	3		Display | SmartPad | Del
Dup	Ctrl	Lt Blue	8	8	Display | SmartPad | Dup
Erase Input	Ctrl	Bksp			Display | SmartPad | Ein
Error Reset or Reset	Esc				Display | SmartPad | Rst

5250 Function	Press these HX2 keys in this order…					RFTerm GUI Equivalent
Exit Program	Green	Ctrl	Lt Blue	9	9	File | Exit
Field Exit	Ctrl	Enter				Display | SmartPad | Fex
Field Minus	Ctrl	Lt Blue	6			Display | SmartPad | F-
Field Plus	Ctrl	Lt Blue	5	5	5	Display | SmartPad | F+
Function Key Editor	Not supported					Session | Configure | FNKeys
Home Key	Not supported					Display | SmartPad | Hm
Help	Ctrl	Lt Blue	4	4		
Insert	Ctrl	Lt Blue	4	4	4	Display | SmartPad | Ins
New Line Key	Ctrl	Lt Blue	6	6		Display | SmartPad | NL
Next or Field Advance or Tab	Tab					Base KeyBar: Tb
Print Screen	Green	Ctrl	Lt Blue	7		Display | SmartPad | Prt
Send or Enter CATV	Enter					Base KeyBar: Ent
Status Line Toggle	Green	Ctrl	Lt Blue	8		Special KeyBar: OIA
SW Rev	Not Supported					File | About
System Request	Ctrl	Lt Blue	7	7	7	7	Display | SmartPad | Srq
Window Down	Ctrl	Down Arrow				Scrollbar
Window Left	Ctrl	Green	Up Arrow			Scrollbar
Window Right	Ctrl	Green	Down Arrow			Scrollbar
Window Up	Ctrl	Up Arrow				Scrollbar
F1 – F9	Green	1 – 9				Display | SmartPad | F1 – F9
F10	Green	0				Display | SmartPad | F10
F11 – F19	Shift	Green	1 – 9			Display | SmartPad | F11 – F19
F20	Shift	Green	0			Display | SmartPad | F20
F21 – F24	Shift	Green	Lt Blue	1 – 4		Display | SmartPad | F21 – F24

MX3X and VX3X Key Maps

Figure A-4 MX3X and VX3X Keypad

The **VX3X keypad** is similar to the MX3X keypad shown above. The VX3X LEDs consist of 2nd, ALT, CTRL, SHFT, CAPS, and STAT. The blue oval label specifies VX3X.

Note: All RFTerm shortcut keys are directly accessible through the RFTerm user interface.

When running RFTerm on any of the MX3X or VX3X devices, please refer to the specific MX3X and VX3X reference guides for inclusive information.

When using a sequence of keys that includes the 2nd key, press the 2nd key first then the rest of the key sequence. Set the On/Off condition of NumLock before pressing a key sequence. There is no visual indication of the condition of NumLock.

Note: When the computer boots, the default condition of NumLock is On.

Note: When the computer boots, the default condition of Caps (or CapsLock) is Off. The Caps (or CapsLock) condition can be toggled with a 2nd-F1 key sequence. The CAPS LED is illuminated when CapsLock is On.

RFTerm Functions – MX3X / VX3X

RFTerm Function	Press these keys and then…			Press this device key	RFTerm GUI Equivalent
	2nd	Ctrl	Shft		
Connect to Host		x	x	C	Session	Connect
Disconnect from Host		x	x	D	Session	Disconnect
Hide / Display All Bars		x	x	L	File	Settings	Display	Hide All Bars
Hide / Display MenuBar		x	x	M	File	Settings	Display	Hide MenuBar
Hide / Display ToolBar		x	x	T	File	Settings	Display	Hide ToolBar
Hide / Display KeyBar		x	x	K	File	Settings	Display	Hide KeyBar
Toggle MenuBar Bottom / Top		x	x	B	File	Settings	Display	Menu at Bottom
Configure Host Session		x	x	N	Session	Configure
Application Settings		x	x	E	File	Settings
Help		x		H	
Increase Font Size		x	x	Up Arrow	ToolBar: A Up Arrow
Decrease Font Size		x	x	Down Arrow	ToolBar: A Down Arrow

RFTerm Function	Press these keys and then…			Press this device key	RFTerm GUI Equivalent
	2nd	Ctrl	Shft		
Toggle Font Bold / Normal		x	x	1	Session | Configure | Font | Font Weight combo box
Display / Hide SmartPad		x	x	P	Display | SmartPad
Display / Hide InputTool		x	x	I (alpha letter i)	Display | InputTool
Scroll Up One Row			x	Up Arrow	Scroll Bar
Scroll Down One Row			x	Down Arrow	Scroll Bar
Scroll Left One Column			x	Left Arrow	Scroll Bar
Scroll Right One Column			x	Right Arrow	Scroll Bar
Exit RFTerm		x	x	X	File | Exit

ANSI Functions – MX3X / VX3X

When running the RFTerm ANSI on these devices, please refer to the **ANSI Plus Reference Guide** for descriptions/explanations of the ANSI functions.

ANSI Function	Press these keys and then…				Press this device key	RFTerm GUI Equivalent
	2nd	Shft	Ctrl	Alt		
Answerback						Display | SmartPad | Ans
Backspace					BkSp	Display | SmartPad | BS
Delete	x				. (period)	Display | SmartPad | Del
Exit Program				x	X	File | Exit
	x	x			X	
Help			x		H	
Print Screen			x	x	P	
Send					Enter	Base KeyBar: Ent
Window Down			x		Down Arrow	Scrollbar or Scroll KeyBar
Window Left			x		Left Arrow	Scrollbar or Scroll KeyBar
Window Right			x		Right Arrow	Scrollbar or Scroll KeyBar
Window Up			x		Up Arrow	Scrollbar or Scroll KeyBar
Local Echo On / Off						Session | Configure | Connection | VT Config | Screen |Local Echo
F1 – F10					F1 – F10	Display | SmartPad | F6 – F10
F11 – F20				x	F1 – F10	Display | SmartPad | F11 – F20
F21 – F30		x			F1 – F10	F17-F24, F25-F32 KeyBars
F31 – F40			x		F1 – F10	F25-F32, F33-F40 KeyBars

IBM 3270 Functions – MX3X / VX3X

When running the RFTerm IBM 3270 terminal emulator on these devices, please refer to the **TN3270 Terminal Emulation Reference Guide** for descriptions/explanations of the 3270 functions.

3270 Function	Press these keys and then…				Press this device key	RFTerm GUI Equivalent
	2nd	Shft	Ctrl	Alt		
Attention			x		A	Display | SmartPad | Attn
Backtab	x				Tab	Base KeyBar: Btb
Clear			x		C	Display | SmartPad | Clr
Cursor Block / Underline						File | Settings | Cursor | Cursor Options
Delete			x		D	Display | SmartPad | Del
Erase					BkSp	Display | SmartPad | BSp
Erase Input			x		BkSp	Display | SmartPad | Ein
Error Reset			x		R	Display | SmartPad | Rst
Exit Program				x	X	File | Exit
		x	x		X	
Help			x		H	
Home Key A Home Key B			x		Left Arrow	Display | SmartPad | Hm
Insert			x		I	Display | SmartPad | Ins
New Line Key			x		N	Display | SmartPad | NL
Next or Tab					Tab	Base KeyBar: Tb
Print Screen			x	x	P	
Roll Up	x				Up Arrow	IBM 3 KeyBar: Roll Up
Roll Down	x				Down Arrow	IBM 3 KeyBar: Roll Dn
SW Rev				x	F then A	File | About
Send					Enter	Base KeyBar: Ent
Status Line Toggle				x	T	Special KeyBar: OIA
		x	x		O	
System Request			x		S	Display | SmartPad | Srq
Window Down			x		Down Arrow	Scrollbar or Scroll KeyBar
Window Right			x		Right Arrow	Scrollbar or Scroll KeyBar
Window Left			x		Left Arrow	Scrollbar or Scroll KeyBar
Window Up			x		Up Arrow	Scrollbar or Scroll KeyBar
PA1 – PA3			x		F1 – F3	Display | SmartPad | Pa1 – Pa3
PF1 – PF10					F1 – F10	Display | SmartPad | F1 – F10
PF11 – PF20				x	F1 – F10	Display | SmartPad | F11 – F20
PF21 – PF24		x			F1 – F4	Display | SmartPad | F21 – F24

IBM 5250 Functions – MX3X / VX3X

When running the RFTerm IBM 5250 terminal emulator on these devices, please refer to the **TN5250 Terminal Emulation Reference Guide** for descriptions/explanations of the 5250 functions.

5250 Function	Press these keys and then…				Press this device key	RFTerm GUI Equivalent
	2nd	Shft	Ctrl	Alt		
Attention			x		A	Display | SmartPad | Attn
Backtab	x				Tab	Base KeyBar: Btb
Char Backspace					BkSp	Display | SmartPad | BSp
Clear			x		C	Display | SmartPad | Clr
Cursor Block / Underline						File | Settings | Cursor | Cursor Options
Delete			x		D	Display | SmartPad | Del
Dup			x		U	Display | SmartPad | Dup
Erase Input			x		BkSp	Display | SmartPad | Ein
Error Reset or Reset					Esc	Base KeyBar: ESC
Exit Program				x	X	File | Exit
		x	x		X	
Help			x		H	
F1 – F10					F1 – F10	Display | SmartPad | F1 – F10
F11 – F20				x	F1 – F10	Display | SmartPad | F11 – F20
F21 – F24		x			F1 – F4	Display | SmartPad | F21 – F24
Field Exit			x		Enter	Display | SmartPad | Fex
					End	
Field Minus			x		M	Display | SmartPad | F-
Field Plus			x		L	Display | SmartPad | F+
Home Key					Home	Display | SmartPad | Hm
Insert			x		I	Display | SmartPad | Ins
New Line Key			x		N	Display | SmartPad | NL
Next or Field Advance or Tab					Tab	Base KeyBar: Tb
Print Screen			x	x	P	Display | SmartPad | Prt
Roll Down	x				Down Arrow	IBM 3 KeyBar: Roll Dn
Roll Up	x				Up Arrow	IBM 3 KeyBar: Roll Up
Send or Enter CATV					Enter	Base KeyBar: Ent
Status Line Toggle				x	T	Special KeyBar: OIA
		x	x		O	

5250 Function	Press these keys and then…				Press this device key	RFTerm GUI Equivalent
	2nd	Shft	Ctrl	Alt		
SW Rev				x	F then A	File | About
System Request			x		S	Display | SmartPad | Srq
Window Down			x		Down Arrow	Scrollbar or Scroll KeyBar
Window Left			x		Left Arrow	Scrollbar or Scroll KeyBar
Window Right			x		Right Arrow	Scrollbar or Scroll KeyBar
Window Up			x		Up Arrow	Scrollbar or Scroll KeyBar

MX5 Key Maps

ANSI/Batch Keypad   IBM 3270 Keypad   IBM 5250 Keypad

Figure A-5 MX5 CE .NET Keypad

Note: All RFTerm shortcut keys are directly accessible through the RFTerm user interface.

When running RFTerm on the MX5, please refer to the MX5 CE .NET Reference Guide for key function information / descriptions.

When using a sequence of keys that includes the Orange or Blue keys, press the color key first then the rest of the key sequence. Ctrl, Alt, Shft, Blue and Orange keys are "sticky keys". They do not need to be held down before pressing the next (or desired) key. It is valid to use combined modifiers on specific keys.

Alphabetic keys default to lower case letters. Press the Shft key, then the alphabetic key for an uppercase letter.

Note: When the computer boots, the default condition of Caps (or CapsLock) is Off. The Caps (or CapsLock) condition can be toggled with Blue plus Tab key sequence.

RFTerm Functions – MX5

RFTerm Function	Press these keys and then…			Press this MX5 key	RFTerm GUI Equivalent
	Orange or Blue	Ctrl	Shft		
Connect to Host		x	x	C	Session | Connect
Disconnect from Host		x	x	D	Session | Disconnect
Hide / Display All Bars		x	x	L	File | Settings | Display | Hide All Bars
Hide / Display MenuBar		x	x	M	File | Settings | Display | Hide MenuBar
Hide / Display ToolBar		x	x	T	File | Settings | Display | Hide ToolBar
Hide / Display KeyBar		x	x	K	File | Settings | Display | Hide KeyBar
Toggle MenuBar Bottom / Top		x	x	B	File | Settings | Display | Menu at Bottom
Configure Host Session		x	x	N	Session | Configure
Application Settings		x	x	E	File | Settings
Increase Font Size		x	x	Up Arrow	ToolBar: A Up Arrow
Decrease Font Size		x	x	Down Arrow	ToolBar: A Down Arrow
Toggle Font Bold / Normal		x	x	1	Session | Configure | Font | Font Weight | combo box
Display / Hide SmartPad		x	x	P	Display | SmartPad
Display / Hide InputTool		x	x	I (alpha letter i)	Display | InputTool
Scroll Up One Row			x	Up Arrow	Scroll Bar
Scroll Down One Row			x	Down Arrow	Scroll Bar
Scroll Left One Row			x	Left Arrow	Scroll Bar
Scroll Right One Row			x	Right Arrow	Scroll Bar
Exit RFTerm		x	x	X	File | Exit

ANSI Functions – MX5

When running the RFTerm ANSI on the MX5, please refer to the **ANSI Plus Reference Guide** for descriptions/explanations of the ANSI functions:

Note: Physical keypad key presses and MX5 virtual keypad presses are the same.

ANSI Function	Press these keys and then…				Press this MX5 key	RFTerm GUI Equivalent
	Orange or Blue	Shft	Ctrl	Alt		
Answerback						Display	SmartPad	Ans
Backspace					BkSp	Display	SmartPad	BS
Delete						Display	SmartPad	Del
Exit Program				x	X	File	Exit
		x	x		X	
Print Screen			x	x	P	
Send					Enter	Base KeyBar: Ent
Window Down			x		Down Arrow	Scrollbar or Scroll KeyBar
Window Left			x		Left Arrow	Scrollbar or Scroll KeyBar
Window Right			x		Right Arrow	Scrollbar or Scroll KeyBar
Window Up			x		Up Arrow	Scrollbar or Scroll KeyBar
Local Echo On / Off						Session	Configure	Connection	VT Config	Screen	Local Echo
F1 – F10	Orange				1 – 10	Display	SmartPad	F1 – F10
F11 – F20	Blue				1 – 10	Display	SmartPad	F11 – F20
F21 – F24	Orange	x			1 – 4	F17-F24 KeyBar
F25 – F30						F25-F32 KeyBar
F31 – F40						F25-F32, F33-F40 KeyBars

IBM 3270 Functions – MX5

When running the RFTerm IBM 3270 terminal emulator on the MX5, please refer to the **TN3270 Terminal Emulation Reference Guide** for descriptions/explanations of the 3270 functions.

MX5 Physical Keypad – 3270

3270 Function	Press these keys and then…				Press this MX5 key	RFTerm GUI Equivalent
	Orange or Blue	Shft	Ctrl	Alt		
Attention			x		A	Display | SmartPad | Attn
Backtab	Orange				Tab	Base KeyBar: Btb
Clear			x		C	Display | SmartPad | Clr
Cursor Block / Underline						File | Settings | Cursor | Cursor Options
Delete			x		D	Display | SmartPad | Del
Erase					BkSp	Display | SmartPad | BSp
Erase Input			x		Q	Display | SmartPad | Ein
Error Reset			x		R	Display | SmartPad | Rst
Exit Program				x	X	File | Exit
		x	x		X	
Home Key A Home Key B	Orange				Left Arrow	Display | SmartPad | Hm
Insert			x		I	Display | SmartPad | Ins
New Line Key			x		N	Display | SmartPad | NL
Next or Tab					Tab	Base KeyBar: Tb
Print Screen			x	x	P	
Roll Up			x		Down Arrow	IBM 3 KeyBar: Roll Up
Roll Down			x		Up Arrow	IBM 3 KeyBar: Roll Dn
SW Rev				x	F then A	File | About
Send					Enter	Base KeyBar: Ent
Status Line Toggle				x	T	Special KeyBar: OIA
		x	x		O	
System Request			x		S	Display | SmartPad | Srq
Window Down			x		Down Arrow	Scrollbar or Scroll KeyBar
Window Right			x		Right Arrow	Scrollbar or Scroll KeyBar
Window Left			x		Left Arrow	Scrollbar or Scroll KeyBar
Window Up			x		Up Arrow	Scrollbar or Scroll KeyBar
PA1 – PA3			x		1 – 3	Display | SmartPad | Pa1 – Pa3

3270 Function	Press these keys and then…				Press this MX5 key	RFTerm GUI Equivalent
	Orange or Blue	Shft	Ctrl	Alt		
PF1 – PF10	Orange				1 – 10	Display | SmartPad | F1 – F10
PF11 – PF20	Blue				1 – 10	Display | SmartPad | F11 – F20
PF21 – PF24	Orange	x			1 – 4	Display | SmartPad | F21 – F24

MX5 Virtual Keypad – 3270

3270 Function	Press these keys and then…		Press this Virtual key	RFTerm GUI Equivalent
	Shft	Ctrl		
Attention		x	A	Display | SmartPad | Attn
Clear		x	C	Display | SmartPad | Clr
Delete		x	D	Display | SmartPad | Del
Erase			Backspace	Display | SmartPad | BSp
Erase Input		x	Backspace	Display | SmartPad | Ein
Error Reset		x	R	Display | SmartPad | Rst
Exit Program	x	x	X	File | Exit
Insert		x	I	Display | SmartPad | Ins
New Line Key		x	N	Display | SmartPad | NL
Next or Tab			Tab	Base KeyBar: Tb
Send			Enter	Base KeyBar: Ent
Status Line Toggle	x	x	O	Special KeyBar: OIA
System Request		x	S	Display | SmartPad | Srq
Window Down		x	Down Arrow	Scrollbar or Scroll KeyBar
Window Right		x	Right Arrow	Scrollbar or Scroll KeyBar
Window Left		x	Left Arrow	Scrollbar or Scroll KeyBar
Window Up		x	Up Arrow	Scrollbar or Scroll KeyBar
PA1 – PA3		x	1 – 3	Display | SmartPad | Pa1 – Pa3

IBM 5250 Functions – MX5

When running the RFTerm IBM 5250 terminal emulator on MX5 devices, please refer to the **TN5250 Terminal Emulation Reference Guide** for descriptions/explanations of the 5250 functions.

MX5 Physical Keypad – 5250

5250 Function	Press these keys and then…				Press this MX5 key	RFTerm GUI Equivalent
	Orange or Blue	Shft	Ctrl	Alt		
Attention			x		A	Display | SmartPad | Attn
Backtab	Orange				Tab	Base KeyBar: Btb
Char Backspace	Orange				Spc	Display | SmartPad | BSp
Clear			x		C	Display | SmartPad | Clr
Cursor Block / Underline						File | Settings | Cursor | Cursor Options
Delete			x		D	Display | SmartPad | Del
Dup			x		U	Display | SmartPad | Dup
Erase Input			x		Q	Display | SmartPad | Ein
Error Reset or Reset					Esc	Base KeyBar: ESC
Exit Program				x	X	File | Exit
		x	x		X	
F1 – F10	Orange				1 – 10	Display | SmartPad | F1 – F10
F11 – F20	Blue				1 – 10	Display | SmartPad | F11 – F20
F21 – F24	Orange	x			1 – 4	Display | SmartPad | F21 – F24
Field Exit			x		Enter	Display | SmartPad | Fex
					Left Scan Button [5]	
Field Minus			x		M	Display | SmartPad | F-
Field Plus			x		L	Display | SmartPad | F+
Home Key	Orange				Left Arrow	Display | SmartPad | Hm
Insert			x		I	Display | SmartPad | Ins
New Line Key			x		N	Display | SmartPad | NL
Next or Field Advance or Tab					Tab	Base KeyBar: Tb
Print Screen			x	x	P	Display | SmartPad | Prt
Roll Down			x		Down Arrow	IBM 3 KeyBar: Roll Dn
Roll Up			x		Up Arrow	IBM 3 KeyBar: Roll Up
Send or Enter CATV					Enter	Base KeyBar: Ent
Status Line Toggle				x	T	Special KeyBar: OIA
		x	x		O	

---

[5] Refer to the "MX5 CE .NET Reference Guide", section titled "Programmable Buttons" for instruction.

5250 Function	Press these keys and then…				Press this MX5 key	RFTerm GUI Equivalent
	Orange or Blue	Shft	Ctrl	Alt		
SW Rev				x	F then A	File | About
System Request			x		S	Display | SmartPad | Srq
Window Down			x		Down Arrow	Scrollbar or Scroll KeyBar
Window Left			x		Left Arrow	Scrollbar or Scroll KeyBar
Window Right			x		Right Arrow	Scrollbar or Scroll KeyBar
Window Up			x		Up Arrow	Scrollbar or Scroll KeyBar

MX5 Virtual Keypad – 5250

5250 Function	Press these keys and then…		Press this Virtual key	RFTerm GUI Equivalent
	Shft	Ctrl		
Attention		x	A	Display | SmartPad | Attn
Clear		x	C	Display | SmartPad | Clr
Delete		x	D	Display | SmartPad | Del
Dup		x	U	Display | SmartPad | Dup
Erase Input		x	Backspace	Display | SmartPad | Ein
Exit Program	x	x	X	File | Exit
Field Exit		x	Enter	Display | SmartPad | Fex
Field Minus		x	M	Display | SmartPad | F-
Field Plus		x	L	Display | SmartPad | F+
Insert		x	I	Display | SmartPad | Ins
New Line Key		x	N	Display | SmartPad | NL
Next or Field Advance or Tab			Tab	Base KeyBar: Tb
Send or Enter CATV			Enter	Base KeyBar: Ent
Status Line Toggle	x	x	O	Special KeyBar: OIA
System Request		x	S	Display | SmartPad | Srq
Window Down		x	Down Arrow	Scrollbar or Scroll KeyBar
Window Left		x	Left Arrow	Scrollbar or Scroll KeyBar
Window Right		x	Right Arrow	Scrollbar or Scroll KeyBar
Window Up		x	Up Arrow	Scrollbar or Scroll KeyBar

MX6/MX6L Key Maps

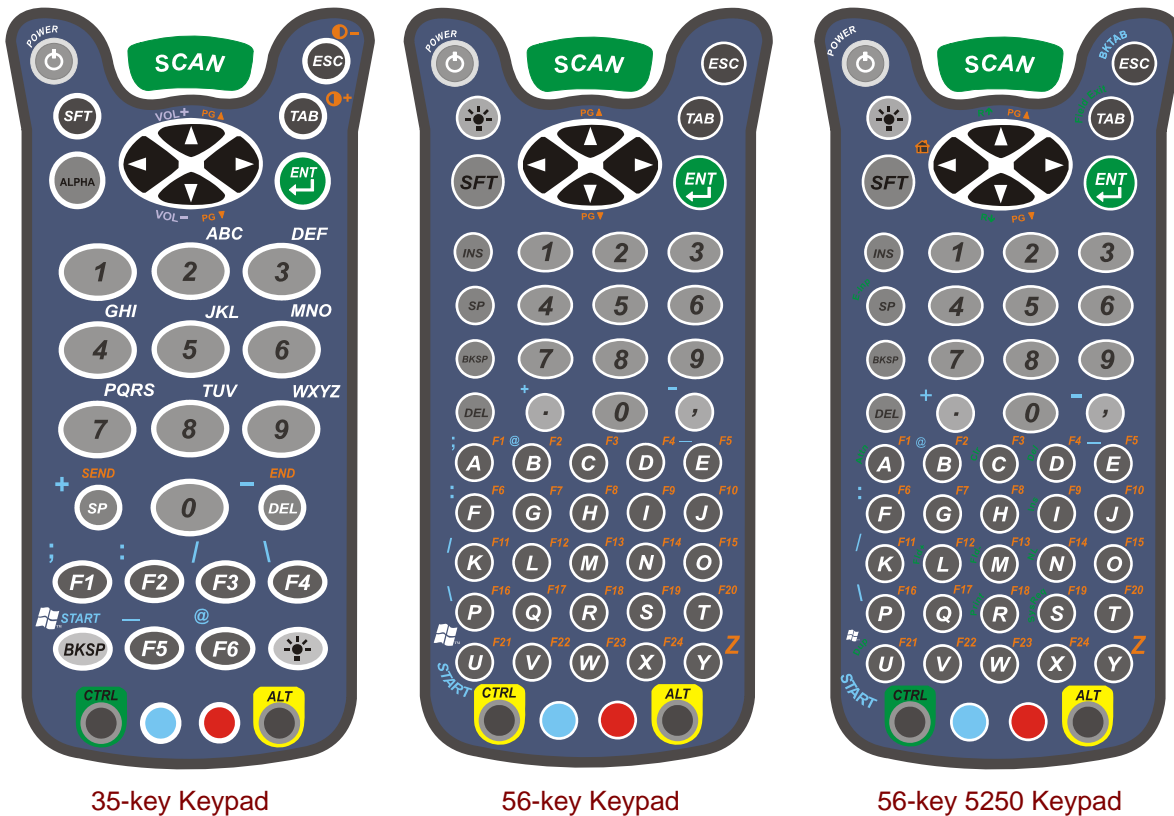| 35-key Keypad | 56-key Keypad | 56-key 5250 Keypad |

Figure A-6 M6 Keypads

Note: All RFTerm shortcut keys are directly accessible through the RFTerm user interface.

When running RFTerm on the MX6, please refer to the MX6 Reference Guide for key function information / descriptions.

When using a sequence of keys that includes the Red or Blue keys, press the color key first then the rest of the key sequence.

Ctrl, Alt, Shft, Blue and Red keys are "sticky keys". They do not need to be held down before pressing the next (or desired) key. It is valid to use combined modifiers on specific keys.

Alphabetic keys default to lower case letters. Press the Shft key, then the alphabetic key for an uppercase letter.

Note: When the computer boots, the default condition of Caps (or CapsLock) is Off. The Caps (or CapsLock) condition can be toggled with Blue plus Tab key sequence.

RFTerm Functions – MX6

56-Key Keypad – MX6

RFTerm Function	Press these keys and then…			Press this MX6 key	RFTerm GUI Equivalent
	Red or Blue	CTRL	SFT		
Connect to Host		x	x	C	Session | Connect
Disconnect from Host		x	x	D	Session | Disconnect
Exit the Emulator				ALT + X	File | Exit
		x	x	X	
Help		x		H	N/A
Hide / Display All Bars		x	x	L	File | Settings | Display | Hide All Bars
Hide / Display MenuBar		x	x	M	File | Settings | Display | Hide MenuBar
Hide / Display ToolBar		x	x	T	File | Settings | Display | Hide ToolBar
Hide / Display KeyBar		x	x	K	File | Settings | Display | Hide KeyBar
Toggle MenuBar Bottom / Top		x	x	B	File | Settings | Display | Menu at Bottom
Configure Host Session		x	x	N	Session | Configure
Application Settings				Alt + E	File | Settings
Increase Font Size		x	x	Up Arrow	Display | Font Up
Decrease Font Size		x	x	Down Arrow	Display | Font Down
Toggle Font Bold / Normal		x	x	1	Session | Configure | Font tab
Display / Hide SmartPad		x	x	P	Display | SmartPad
Display / Hide InputTool		x	x	I (alpha letter i)	Display | InputTool
Scroll Up One Row			x	Up Arrow	Scroll Bar
Scroll Down One Row			x	Down Arrow	Scroll Bar
Scroll Left One Row			x	Left Arrow	Scroll Bar
Scroll Right One Row			x	Right Arrow	Scroll Bar

35-Key Keypad – MX6

RFTerm Function	Press these keys and then…			Press this MX6 key	RFTerm GUI Equivalent
	Red or Blue	CTRL	SFT		
Connect to Host	Alpha	x	x	2 2 2	Session | Connect
Disconnect from Host	Alpha	x	x	3	Session | Disconnect
Exit the Emulator	Alpha			ALT + 9 9	File | Exit
Hide / Display All Bars	Alpha	x	x	5 5 5	File | Settings | Display | Hide All Bars
Hide / Display MenuBar	Alpha	x	x	6	File | Settings | Display | Hide MenuBar
Hide / Display ToolBar	Alpha	x	x	8	File | Settings | Display | Hide ToolBar
Hide / Display KeyBar	Alpha	x	x	5 5	File | Settings | Display | Hide KeyBar
Toggle MenuBar Bottom / Top	Alpha	x	x	2 2	File | Settings | Display | Menu at Bottom
Configure Host Session	Alpha	x	x	6 6	Session | Configure
Application Settings	Alpha			ALT + 3 3	File | Settings
Increase Font Size		x	x	Up Arrow	Display | Font Up
Decrease Font Size		x	x	Down Arrow	Display | Font Down
Toggle Font Bold / Normal		x	x	1	Session | Configure | Font
Display / Hide SmartPad	Alpha	x	x	7	Display | SmartPad
Display / Hide InputTool	Alpha	x	x	4 4 4	Display | InputTool
Help	Alpha	x		4 4	
Scroll Up One Row			x	Up Arrow	Scroll Bar
Scroll Down One Row			x	Down Arrow	Scroll Bar
Scroll Left One Row			x	Left Arrow	Scroll Bar
Scroll Right One Row			x	Right Arrow	Scroll Bar

ANSI Functions – MX6

When running the RFTerm ANSI on the MX6, please refer to the **ANSI Plus Reference Guide** for descriptions/explanations of the ANSI functions:

Note: Physical keypad key presses and MX6 virtual keypad presses are the same.

MX6 56-Key Keypad – ANSI

ANSI Function	Press these keys and then…				Press this MX6 key	RFTerm GUI Equivalent
	Red or Blue	SFT	CTRL	ALT		
Answerback						Display | SmartPad | Ans
Backspace					BKSP	Display | SmartPad | BS
Delete					DEL	Display | SmartPad | Del
Exit Program [6]				x	X	File | Exit
		x	x		X	
Help			x		H	
Local Echo On / Off						Session | Configure | Connection | VT Config | Screen | Local Echo
Print Screen			x	x	P	
Send					ENT	Display | Smartpad | Ent
Window Down			x		Down Arrow	Scrollbar
Window Left			x		Left Arrow	Scrollbar
Window Right			x		Right Arrow	Scrollbar
Window Up			x		Up Arrow	Scrollbar
F1 – F10	Red				A – J	Display | SmartPad | F1 – F10
F11 – F20	Red				K – T	Display | SmartPad | F11 – F20
F21 – F30	Red				U – X	KeyBar
F31 – F40						KeyBar

---

[6] The MX6 Virtual Keypad accepts only Shift+Ctrl+X as an Exit Program command.

MX6 35-Key Keypad – ANSI

ANSI Function	Press these keys and then…				Press this MX6 key	RFTerm GUI Equivalent
	Red or Blue	SFT	CTRL	ALT		
Answerback						Display | SmartPad | Ans
Backspace					BKSP	Display | SmartPad | BS
Delete					DEL	Display | SmartPad | Del
Exit Program [7]	Alpha			x	9 9	File | Exit
Function Key Editor					**	Session | Configure | FNKeys
Help	Alpha		x		4 4	
Print Screen	Alpha		x	x	7	
Send					ENT	Display | Smartpad | Ent
Window Down			x		Down Arrow	Scrollbar
Window Left			x		Left Arrow	Scrollbar
Window Right			x		Right Arrow	Scrollbar
Window Up			x		Up Arrow	Scrollbar
F1 – F6 and F7 – F10			x		F1 – F4	Display | SmartPad | F1 – F10
F11 – F16 or F11 – F12			x (F11–F12 row)	x (F11–F16 row)	F1 – F6 / F5 – F6	Display | SmartPad | F11 – F20
F21 – F26		x			F1 – F6	KeyBar
F27 – F40					**	KeyBar

** Not Supported

---

[7] The MX6 Virtual Keypad accepts only Shift+Ctrl+X as an Exit Program command.

IBM 3270 Functions – MX6

When running the RFTerm IBM 3270 terminal emulator on the MX6, please refer to the **TN3270 Terminal Emulation Reference Guide** for descriptions/explanations of the 3270 functions.

MX6 56-Key Physical Keypad – 3270

3270 Function	Press these keys and then…				Press this MX6 key	RFTerm GUI Equivalent
	Red or Blue	SFT	CTRL	ALT		
Attention			x		A	Display	SmartPad	Attn
Backtab		x			TAB	Display	SmartPad	Btb
Clear			x		C	Display	SmartPad	Clr
Cursor Block / Underline						File	Settings	Cursor	Cursor Options
Delete			x		D	Display	SmartPad	Del
Erase					BKSP	Display	SmartPad	BSp
Erase Input			x		BKSP	Display	SmartPad	Ein
Error Reset			x		R	Display	SmartPad	Rst
Exit Program				x	X	File	Exit
		x	x		X	
Function Key Editor					**	Session	Configure	FNKeys
Help			x		H	
Home Key					**	Display	SmartPad	Hm
Insert			x		I	Display	SmartPad	Ins
New Line Key			x		N	Display	SmartPad	NL
Next or Tab					TAB	Display	SmartPad	Tb
Print Screen			x	x	P	
Roll Up			x		Up Arrow	IBM 3 KeyBar: Roll Up
Roll Down			x		Down Arrow	IBM 3 KeyBar: Roll Dn
SW Rev					**	File	About
Send					ENT	Base KeyBar: Ent
Status Line Toggle				x	T	File	Settings	Display	Hide OIA
		x	x		O	
System Request			x		S	Display	SmartPad	Srq
Window Down			x		Down Arrow	Scrollbar
Window Right			x		Right Arrow	Scrollbar
Window Left			x		Left Arrow	Scrollbar
Window Up			x		Up Arrow	Scrollbar

3270 Function	Press these keys and then…				Press this MX6 key	RFTerm GUI Equivalent
	Red or Blue	SFT	CTRL	ALT		
PA1 – PA3			x		1 – 3	Display | SmartPad | Pa1 – Pa3
PF1 – PF10	Red				A – J	Display | SmartPad | F1 – F10
PF11 – PF20	Red				K – T	Display | SmartPad | F11 – F20
PF21 – PF24	Red				U – X	Display | SmartPad | F21 – F24

** Not Supported

MX6 35-Key Physical Keypad – 3270

3270 Function	Press these keys and then…				Press this MX6 key	RFTerm GUI Equivalent
	Red or Blue	SFT	CTRL	ALT		
Backtab	Blue				TAB	Base KeyBar: Btb
Exit Program						File | Exit
Home Key	Red				Left Arrow	Display | SmartPad | Hm
Next or Tab					TAB	Base KeyBar: Tb
Roll Up			x		Up Arrow	IBM 3 KeyBar: Roll Up
Roll Down			x		Down Arrow	IBM 3 KeyBar: Roll Dn
Send					ENT	Base KeyBar: Ent
Window Down			x		Down Arrow	Scrollbar or Scroll KeyBar
Window Right			x		Right Arrow	Scrollbar or Scroll KeyBar
Window Left			x		Left Arrow	Scrollbar or Scroll KeyBar
Window Up			x		Up Arrow	Scrollbar or Scroll KeyBar
PA1 – PA3			x		1 – 3	Display | SmartPad | Pa1 – Pa3
F1 – F6					F1 - F6	Display | SmartPad | F1 – F6
F7 – F10			x		F1 – F4	Display | SmartPad | F7 – F10
F11			or x	x	F1 or F5	Display | SmartPad | F11
F12			or x	x	F2 or F6	Display | SmartPad | F12
F13 – F16				x	F3 – F6	Display | SmartPad | F13 – F16

3270 Function	Press these keys and then…				Press this MX6 key	RFTerm GUI Equivalent
	Red or Blue	SFT	CTRL	ALT		
F21 – F26		x			F1 – F6	Display | SmartPad | F21 – F26

MX6 Virtual Keypad - 3270

3270 Function	Press these keys and then…		Press this Virtual key	RFTerm GUI Equivalent
	Shift	Ctl		
Attention		x	A	Display | SmartPad | Attn
Backtab	x		Tab	Display | SmartPad | Btb
Clear		x	C	Display | SmartPad | Clr
Delete		x	D	Display | SmartPad | Del
Erase			Backspace	Display | SmartPad | BSp
Erase Input		x	Backspace	Display | SmartPad | Ein
Error Reset		x	R	Display | SmartPad | Rst
Exit Program	x	x	X	File | Exit
Function Key Editor			**	Session | Configure | FNKeys
Insert		x	I	Display | SmartPad | Ins
New Line Key		x	N	Display | SmartPad | NL
Next or Tab			Tab	Base KeyBar: Tb
Print Screen		x	ALT + P	
Roll Up		x	Up Arrow	IBM 3 KeyBar: Roll Up
Roll Down		x	Down Arrow	IBM 3 KeyBar: Roll Dn
Send			Enter	Display | SmartPad | Ent
Status Line Toggle			ALT + T	Special KeyBar: OIA
System Request		x	S	Display | SmartPad | Srq
Window Down		x	Down Arrow	Scrollbar
Window Right		x	Right Arrow	Scrollbar
Window Left	x		Left Arrow	Scrollbar
Window Up		x	Up Arrow	Scrollbar
PA1 - PA3		x	1 - 3	Display | SmartPad | Pa1 - Pa3

** Not Supported

IBM 5250 Functions – MX6

When running the RFTerm IBM 5250 terminal emulator on MX6 devices, please refer to the **TN5250 Terminal Emulation Reference Guide** for descriptions/explanations of the 5250 functions.

MX6 56-Key Physical Keypad - 5250

5250 Function	Red or Blue	SFT	CTRL	ALT	Press this MX6 key	RFTerm GUI Equivalent
Attention			x		A	Display | SmartPad | Attn
Backtab		x			TAB	Base KeyBar: Btb
Char Backspace					BKSP	Display | SmartPad | BSp
Clear			x		C	Display | SmartPad | Clr
Cursor Block / Underline						File | Settings | Cursor | Cursor Options
Delete			x		D	Display | SmartPad | Del
Dup			x		U	Display | SmartPad | Dup
Erase Input			x		BKSP	Display | SmartPad | Ein
Error Reset or Reset					ESC	Base KeyBar: ESC
Exit Program				x	X	File | Exit
		x	x		X	
F1 – F10	Red				A – J	Display | SmartPad | F1 – F10
F11 – F20	Red				K – T	Display | SmartPad | F11 – F20
F21 – F24	Red				U – X	Display | SmartPad | F21 – F24
Field Exit			x		ENT	Display | SmartPad | Fex
Field Minus			x		M	Display | SmartPad | F-
Field Plus			x		L	Display | SmartPad | F+
Home Key						Display | SmartPad | Hm
Help			x		H	
Insert			x		I	Display | SmartPad | Ins
New Line Key			x		N	Display | SmartPad | NL
Next or Field Advance or Tab					Tab	Base KeyBar: Tb
Print Screen			x	x	P	Display | SmartPad | Prt
Roll Down			x		Down Arrow	IBM 3 KeyBar: Roll Dn
Roll Up			x		Up Arrow	IBM 3 KeyBar: Roll Up
Send or Enter CATV					ENT	Base KeyBar: Ent

5250 Function	Press these keys and then…				Press this MX6 key	RFTerm GUI Equivalent
	Red or Blue	SFT	CTRL	ALT		
Status Line Toggle				x	T	Special KeyBar: OIA
		x	x		O	
SW Rev					**	File | About
System Request			x		S	Display | SmartPad | Srq
Window Down			x		Down Arrow	Scrollbar
Window Left			x		Left Arrow	Scrollbar
Window Right			x		Right Arrow	Scrollbar
Window Up			x		Up Arrow	Scrollbar

** Not Supported

MX6 35-Key Physical Keypad - 5250

5250 Function	Press these keys and then…				Press this MX6 key	RFTerm GUI Equivalent
	Red or Blue	SFT	CTRL	ALT		
Attention	Alpha		x		1	Display | SmartPad | Attn
Backtab		x			TAB	Display | SmartPad | Btb
Char Backspace					BKSP	Display | SmartPad | BSp
Clear	Alpha		x		2 2 2	Display | SmartPad | Clr
Delete	Alpha		x		3	Display | SmartPad | Del
Dup	Alpha		x		8 8	Display | SmartPad | Dup
Erase Input			x		BKSP	Display | SmartPad | Eln
Error Reset or Reset					ESC	Display | SmartPad | Rst
Exit Program	Alpha			x	9 9	File | Exit
Field Exit			x		ENT	Display | SmartPad | FEx
Field Minus	Alpha		x		6	Display | SmartPad | F-
Field Plus	Alpha		x		5 5 5	Display | SmartPad | F+
Function Key Editor					**	Session | Configure | FNKeys
Help	Alpha		x		4 4	
Home Key					N/A	Display | SmartPad | Hm
Insert	Alpha		x		4 4 4	Display | SmartPad | Ins
New Line Key	Alpha		x		6 6	Display | SmartPad | Nl
Next or Field Advance or Tab					TAB	Base KeyBar: Tb
Print Screen	Alpha		x	x	7	Display | SmartPad | Prt

5250 Function	Press these keys and then…				Press this MX6 key	RFTerm GUI Equivalent
	Red or Blue	SFT	CTRL	ALT		
Status Line Toggle	Alpha			x	8	File	settings	Display	Hide O1A
Sys_Req	Alpha		x		7 7 7 7	Display	SmartPad	Srq
SW Rev					**	File	About
Send or Enter CATV					ENT	Display	SmartPad	Ent
Window Down			x		Down Arrow	Scrollbar
Window Left			x		Left Arrow	Scrollbar
Window Right			x		Right Arrow	Scrollbar
Window Up			x		Up Arrow	Scrollbar
F1 – F6 and F7 – F10			x		F1 – F4	Display	SmartPad	F1 – F10
F11 – F16 or F11 – F12				x	F1 – F6	Display	SmartPad	F11 – F20
			x		F5 – F6	
F21 – F26		x			F1 – F6	Display	SmartPad	F21 – F24

**   Not Supported

MX6 Virtual Keypad - 5250

5250 Function	Press these keys and then…		Press this Virtual key	RFTerm GUI Equivalent
	Shift	Ctl		
Attention		x	A	Display | SmartPad | Attn
BackTab	x		Tab	Display | SmartPad | BTb
Char Backspace			Backspace	Display | SmartPad | BSp
Clear		x	C	Display | SmartPad | Clr
Delete		x	D	Display | SmartPad | Del
Dup		x	U	Display | SmartPad | Dup
Erase Input		x	Backspace	Display | SmartPad | EIn
Error Reset or Reset	Blue		ALT	Display | SmartPad | Rst
Exit Program			ALT + X	File | Exit
Field Exit		x	Enter	Display | SmartPad | FEx
Field Minus		x	M	Display | SmartPad | F-
Field Plus		x	L	Display | SmartPad | F+
Insert		x	I	Display | SmartPad | Ins
New Line Key		x	N	Display | SmartPad | NL
Next or Field Advance or Tab			Tab	Display | SmartPad | Tb
Print Screen		x	Alt + P	Display | SmartPad | Prt
Send or Enter CATV			Enter	Base KeyBar: Ent
Status Line Toggle			**	File | Settings | Display | Hide OIA
System Request		x	S	Display | SmartPad | Srq
Window Down		x	Down Arrow	Scrollbar or Scroll KeyBar
Window Left		x	Left Arrow	Scrollbar or Scroll KeyBar
Window Right		x	Right Arrow	Scrollbar or Scroll KeyBar
Window Up		x	Up Arrow	Scrollbar or Scroll KeyBar

** Not Supported

MX7 Key Maps
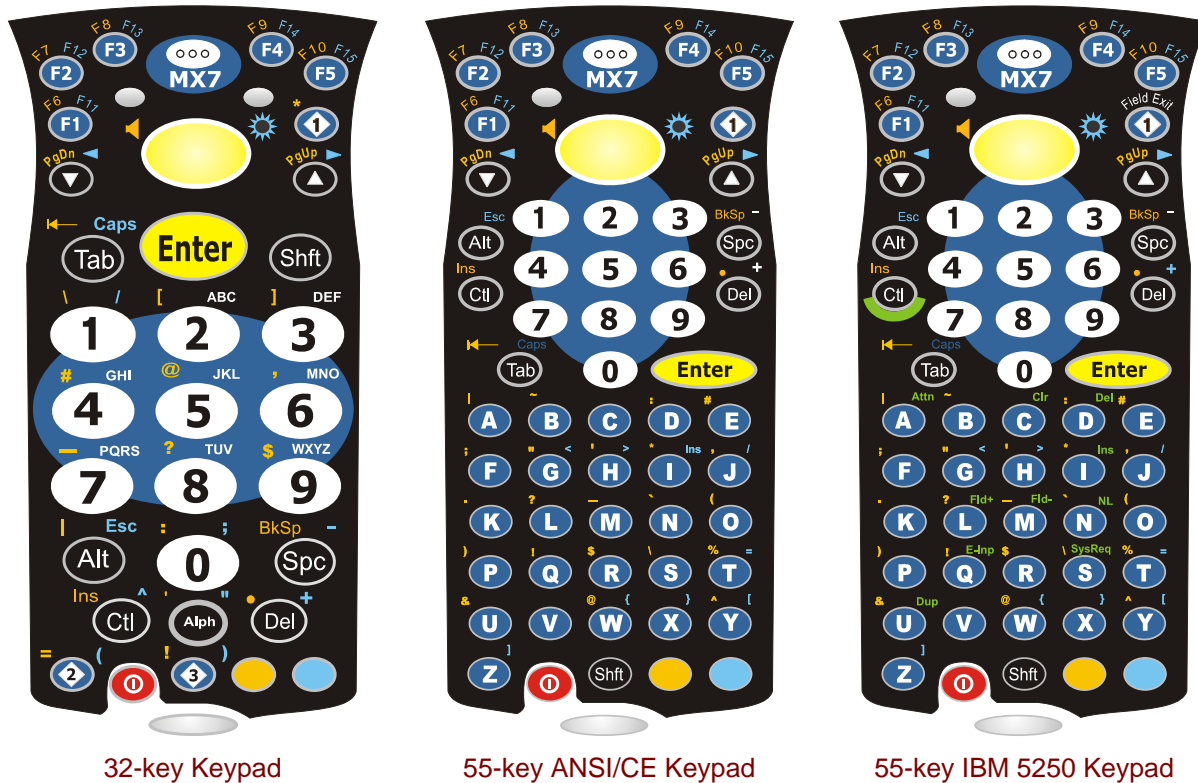
| 32-key Keypad | 55-key ANSI/CE Keypad | 55-key IBM 5250 Keypad |

Figure A-7 MX7 Keypads

Note: All RFTerm shortcut keys are directly accessible through the RFTerm user interface.

When running RFTerm on the MX7, please refer to the MX7 Reference Guide for key function information / descriptions.

When using a sequence of keys that includes the Orange or Blue keys, press the color key first then the rest of the key sequence. Ctrl, Alt, Shft, Blue and Orange keys are "sticky keys". They do not need to be held down before pressing the next (or desired) key. It is valid to use combined modifiers on specific keys.

Alphabetic keys default to lower case letters. Press the Shft key, then the alphabetic key for an uppercase letter.

Note: When the computer boots, the default condition of Caps (or CapsLock) is Off. The Caps (or CapsLock) condition can be toggled with Blue plus Tab key sequence.

RFTerm Functions – MX7

55-Key Keypad – MX7

RFTerm Function	Press these keys and then…			Press this MX7 key	RFTerm GUI Equivalent
	Orange or Blue	Ctl	Shft		
Connect to Host		x	x	C	Session | Connect
Disconnect from Host		x	x	D	Session | Disconnect
Exit the Emulator				ALT + X	File | Exit
		x	x	X	
Help		x		H	N/A
Hide / Display All Bars		x	x	L	File | Settings | Display | Hide All Bars
Hide / Display MenuBar		x	x	M	File | Settings | Display | Hide MenuBar
Hide / Display ToolBar		x	x	T	File | Settings | Display | Hide ToolBar
Hide / Display KeyBar		x	x	K	File | Settings | Display | Hide KeyBar
Toggle MenuBar Bottom / Top		x	x	B	File | Settings | Display | Menu at Bottom
Configure Host Session		x	x	N	Session | Configure
Application Settings		x	x	E	File | Settings
Increase Font Size		x	x	Up Arrow	Display | Font Up
Decrease Font Size		x	x	Down Arrow	Display | Font Down
Toggle Font Bold / Normal		x	x	1	Session | Configure | Font tab
Display / Hide SmartPad		x	x	P	Display | SmartPad
Display / Hide InputTool		x	x	I (alpha letter i)	Display | InputTool
Scroll Up One Row			x	Up Arrow	Scroll Bar
Scroll Down One Row			x	Down Arrow	Scroll Bar
Scroll Left One Row	Blue		x	Down Arrow	Scroll Bar
Scroll Right One Row	Blue		x	Up Arrow	Scroll Bar

32-Key Keypad – MX7

RFTerm Function	Press these keys and then…			Press this MX7 key	RFTerm GUI Equivalent
	Orange or Blue	Ctl	Shft		
Connect to Host	Alpha	x	x	2 2 2	Session | Connect
Disconnect from Host	Alpha	x	x	3	Session | Disconnect
Exit the Emulator	Alpha			ALT + 9 9	File | Exit
Hide / Display All Bars	Alpha	x	x	5 5 5	File | Settings | Display | Hide All Bars
Hide / Display MenuBar	Alpha	x	x	6	File | Settings | Display | Hide MenuBar
Hide / Display ToolBar	Alpha	x	x	8	File | Settings | Display | Hide ToolBar
Hide / Display KeyBar	Alpha	x	x	5 5	File | Settings | Display | Hide KeyBar
Toggle MenuBar Bottom / Top	Alpha	x	x	2 2	File | Settings | Display | Menu at Bottom
Configure Host Session	Alpha	x	x	6 6	Session | Configure
Application Settings	Alpha	x	x	3 3	File | Settings
Increase Font Size		x	x	Up Arrow	Display | Font Up
Decrease Font Size		x	x	Down Arrow	Display | Font Down
Toggle Font Bold / Normal		x	x	1	Session | Configure | Font
Display / Hide SmartPad	Alpha	x	x	7	Display | SmartPad
Display / Hide InputTool	Alpha	x	x	4 4 4	Display | InputTool
Help	Alpha	x		4 4	
Scroll Up One Row			x	Up Arrow	Scroll Bar
Scroll Down One Row			x	Down Arrow	Scroll Bar
Scroll Left One Row	Blue		x	Down Arrow	Scroll Bar
Scroll Right One Row	Blue		x	Up Arrow	Scroll Bar

ANSI Functions – MX7

When running the RFTerm ANSI on the MX7, please refer to the **ANSI Plus Reference Guide** for descriptions/explanations of the ANSI functions:

Note: Physical keypad key presses and MX7 virtual keypad presses are the same.

MX7 55-Key Keypad – ANSI

ANSI Function	Press these keys and then…				Press this MX7 key	RFTerm GUI Equivalent
	Orange or Blue	Shft	Ctl	Alt		
Answerback					**	Display | SmartPad | Ans
Backspace	Orange				Space	Display | SmartPad | BS
Delete					Del	Display | SmartPad | Del
Exit Program				x	X	File | Exit
		x	x		X	
Function Key Editor					**	Session | Configure | FNKeys
Help			x		H	
Local Echo On / Off						Session | Configure | Connection | VT Config | Screen | Local Echo
Print Screen			x	x	P	
Send					Enter	Display | Smartpad | Ent
Window Down			x		Down Arrow	Scrollbar
Window Left	Blue		x		Down Arrow	Scrollbar
Window Right	Blue		x		Up Arrow	Scrollbar
Window Up			x		Up Arrow	Scrollbar
F1 – F5 and F6 – F10	Orange				F1 – F5	Display | SmartPad | F1 – F10
F11 – F15 F16 – F20	Blue	x			F1 – F5	Display | SmartPad | F11 – F20
F21 – F24	Orange		x		F1 – F4	KeyBar
F25 – F40					**	KeyBar

**   Not Supported

MX7 32-Key Keypad – ANSI

| ANSI Function | Press these keys and then… | | | | Press this MX7 key | RFTerm GUI Equivalent |
	Orange or Blue	Shft	Ctl	Alt		
Answerback					**	Display | SmartPad | Ans
Backspace	Orange				BKSP	Display | SmartPad | BS
Delete					DEL	Display | SmartPad | Del
Exit Program [8]	Alpha			x	9 9	File | Exit
Function Key Editor					**	Session | Configure | FNKeys
Help	Alpha		x		4 4	
Print Screen	Alpha		x	x	7	
Send					ENT	Display | Smartpad | Ent
Window Down			x		Down Arrow	Scrollbar
Window Left	Blue		x		Down Arrow	Scrollbar
Window Right	Blue		x		Up Arrow	Scrollbar
Window Up			x		Up Arrow	Scrollbar
F1 – F5 and F6 – F10	Orange				F1 – F5	Display | SmartPad | F1 – F10
F11 – F15	Blue				F1 – F5	Display | SmartPad |
F16 – F20		x			F1 – F5	F11 – F20
F21 – F24	Orange		x		F1 – F4	KeyBar
F25 – F40					**	KeyBar

**   Not Supported

---

[8] The MX7 Virtual Keypad accepts only Shift+Ctrl+X as an Exit Program command.

IBM 3270 Functions – MX7

When running the RFTerm IBM 3270 terminal emulator on the MX7, please refer to the **TN3270 Terminal Emulation Reference Guide** for descriptions/explanations of the 3270 functions.

MX7 55-Key Physical Keypad – 3270

3270 Function	Press these keys and then…				Press this MX7 key	RFTerm GUI Equivalent
	Orange or Blue	Shft	Ctl	Alt		
Attention			x		A	Display | SmartPad | Attn
Backtab	Orange				TAB	Display | SmartPad | Btb
Clear			x		C	Display | SmartPad | Clr
Cursor Block / Underline						File | Settings | Cursor | Cursor Options
Delete			x		D	Display | SmartPad | Del
Erase	Orange				Spc	Display | SmartPad | BSp
Erase Input	Orange		x		Spc	Display | SmartPad | Ein
Error Reset			x		R	Display | SmartPad | Rst
Exit Program				x	X	File | Exit
		x	x		X	
Function Key Editor					**	Session | Configure | FNKeys
Help			x		H	
Home Key					**	Display | SmartPad | Hm
Insert			x		I	Display | SmartPad | Ins
New Line Key			x		N	Display | SmartPad | NL
Next or Tab					Tab	Display | SmartPad | Tb
Print Screen			x	x	P	
Roll Up			x		Up Arrow	IBM 3 KeyBar: Roll Up
Roll Down			x		Down Arrow	IBM 3 KeyBar: Roll Dn
SW Rev					**	File | About
Send					ENT	Base KeyBar: Ent
Status Line Toggle				x	T	File | Settings | Display | Hide OIA
		x	x		O	
System Request			x		S	Display | SmartPad | Srq
Window Down			x		Down Arrow	Scrollbar
Window Right	Blue		x		Up Arrow	Scrollbar
Window Left	Blue		x		Down Arrow	Scrollbar
Window Up			x		Up Arrow	Scrollbar

3270 Function	Press these keys and then…				Press this MX7 key	RFTerm GUI Equivalent
	Orange or Blue	Shft	Ctl	Alt		
PA1 – PA3			x		1 – 3	Display | SmartPad | Pa1 – Pa3
F1 – F5					F1 – F5	Display | SmartPad | F1 – F10
F6 - F10	Orange					
F11 – F15	Blue				F1 – F5	Display | SmartPad | F11 – F20
F16 – F20		x				
F21 – F24 *	Orange		x		F1 – F4	Display | SmartPad | F21 – F24

* F21-F24, press Ctrl first then the Orange key then the Function key.
**   Not Supported

MX7 32-Key Physical Keypad – 3270

3270 Function	Press these keys and then…				Press this MX7 key	RFTerm GUI Equivalent
	Orange or Blue	Shft	Ctl	Alt		
Attention	Alpha		x		2	Display | SmartPad | Attn
Backtab	Orange				Tab	Display | SmartPad | BTb
Clear	Alpha		x		2 2 2	Display | SmartPad | Clr
Delete	Alpha		x		3	Display | SmartPad | Del
Erase	Orange				Spc	Display | SmartPad | Bsp
Erase Input	Orange		x		Spc	Display | SmartPad | Eln
Error Reset or Reset	Alpha		x		7 7 7	Display | SmartPad | Rst
Exit Program	Alpha			x	9	File | Exit
Function Key Editor					**	Session | Configure | FNKeys
Help	Alpha		x		4 4	
Home Key					**	Display | SmartPad | Hm
Insert	Alpha		x		4 4 4	Display | SmartPad | Ins
New Line Key	Alpha		x		6 6	Display | SmartPad | Nl
Next or Tab					Tab	Display | SmartPad | Tb
Print Screen	Alpha		x	x	7	
Status Line Toggle	Alpha			x	8	File | Settings | Display | Hide O1A
SW Rev						File | About
Sys_Req	Alpha		x		7 7 7 7	Display | SmartPad | Srq
Window Down			x		Down Arrow	Scrollbar
Window Right	Blue		x		Up Arrow	Scrollbar
Window Left	Blue		x		Down Arrow	Scrollbar
Window Up			x		Up Arrow	Scrollbar
PA1 – PA3			x		1 – 3	Display | SmartPad | Pa1 – Pa3
F1 – F5					F1 – F5	Display | SmartPad | F1 – F10
F6 – F10	Orange					
F11 – F15	Blue				F1 – F5	Display | SmartPad | F11 – F20
F16 – F20		x				
F21 – F24	Orange		x		F1 – F4	Display | SmartPad | F21 – F24

**   Not Supported

IBM 5250 Functions – MX7

When running the RFTerm IBM 5250 terminal emulator on MX7 devices, please refer to the **TN5250 Terminal Emulation Reference Guide** for descriptions/explanations of the 5250 functions.

MX7 55-Key Physical Keypad - 5250

5250 Function	Press these keys and then…				Press this MX7 key	RFTerm GUI Equivalent
	Orange or Blue	Shft	Ctl	Alt		
Attention			x		A	Display | SmartPad | Attn
Backtab	Orange				Tab	Base KeyBar: Btb
Char Backspace	Orange				Spc	Display | SmartPad | BSp
Clear			x		C	Display | SmartPad | Clr
Cursor Block / Underline						File | Settings | Cursor | Cursor Options
Delete			x		D	Display | SmartPad | Del
Dup			x		U	Display | SmartPad | Dup
Erase Input	Orange		x		Spc	Display | SmartPad | Ein
Error Reset or Reset	Blue			x		Display | SmartPad | Rst
Exit Program				x	X	File | Exit
		x	x		X	
Field Exit			x		Enter	Display | SmartPad | Fex
					Diamond 1	
Field Minus			x		M	Display | SmartPad | F-
Field Plus			x		L	Display | SmartPad | F+
Function Key Editor					**	Session | Configure | FNKeys
Home Key					**	Display | SmartPad | Hm
Help			x		H	
Insert			x		I	Display | SmartPad | Ins
New Line Key			x		N	Display | SmartPad | NL
Next or Field Advance or Tab					Tab	Base KeyBar: Tb
Print Screen			x	x	P	Display | SmartPad | Prt
Send or Enter CATV					Enter	Base KeyBar: Ent
Status Line Toggle				x	T	Special KeyBar: OIA
		x	x		O	
SW Rev					**	File | About
System Request			x		S	Display | SmartPad | Srq
Window Down			x		Down Arrow	Scrollbar

5250 Function	Press these keys and then…				Press this MX7 key	RFTerm GUI Equivalent
	Orange or Blue	Shft	Ctl	Alt		
Window Left	Blue		x		Down Arrow	Scrollbar
Window Right	Blue		x		Up Arrow	Scrollbar
Window Up			x		Up Arrow	Scrollbar
F1 – F5					F1 – F5	Display | SmartPad | F1 – F10
F6 – F10	Orange					
F11 – F15	Blue				F1 – F5	Display | SmartPad | F11 – F20
F16 – F20		x				
F21 – F24 *	Orange		x		F1 – F4	Display | SmartPad | F21 – F24

* F21-F24, press Ctrl first then the Orange key then the Function key.

** Not Supported

MX7 32-Key Physical Keypad - 5250

5250 Function	Press these keys and then…				Press this MX7 key	RFTerm GUI Equivalent
	Orange or Blue	Shft	Ctl	Alt		
Attention	Alpha		x		2	Display | SmartPad | Attn
Backtab	Orange				Tab	Display | SmartPad | Btb
Char Backspace	Orange				Spc	Display | SmartPad | BSp
Clear	Alpha		x		2 2 2	Display | SmartPad | Clr
Delete	Alpha		x		3	Display | SmartPad | Del
Dup	Alpha		x		8 8	Display | SmartPad | Dup
Erase Input	Orange		x		Spc	Display | SmartPad | EIn
Error Reset or Reset	Blue			x		Display | SmartPad | Rst
Exit Program	Alpha			x	9 9	File | Exit
Field Exit			x		Enter	Display | SmartPad | FEx
					Diamond 1	
Field Minus	Alpha		x		6	Display | SmartPad | F-
Field Plus	Alpha		x		5 5 5	Display | SmartPad | F+
Function Key Editor					**	Session | Configure | FNKeys
Help	Alpha		x		4 4	
Home Key					**	Display | SmartPad | Hm
Insert	Alpha		x		4 4 4	Display | SmartPad | Ins
New Line Key	Alpha		x		6 6	Display | SmartPad | Nl

5250 Function	Press these keys and then…				Press this MX7 key	RFTerm GUI Equivalent
	Orange or Blue	Shft	Ctl	Alt		
Next or Field Advance or Tab					Tab	Base KeyBar: Tb
Print Screen	Alpha		x	x	7	Display | SmartPad | Prt
Status Line Toggle	Alpha			x	8	File | settings | Display | Hide O1A
Sys_Req	Alpha		x		7 7 7 7	Display | SmartPad | Srq
SW Rev					**	File | About
Send or Enter CATV					Enter	Display | SmartPad | Ent
Window Down			x		Down Arrow	Scrollbar
Window Left	Blue		x		Down Arrow	Scrollbar
Window Right	Blue		x		Up Arrow	Scrollbar
Window Up			x		Up Arrow	Scrollbar
F1 – F5					F1 – F5	Display | SmartPad | F1 – F10
F6 – F10	Orange					
F11 – F15	Blue				F1 – F5	Display | SmartPad | F11 – F20
F16 – F20		x				
F21 – F24	Orange		x		F1 – F4	Display | SmartPad | F21 – F24

**   Not Supported

MX8 Key Maps
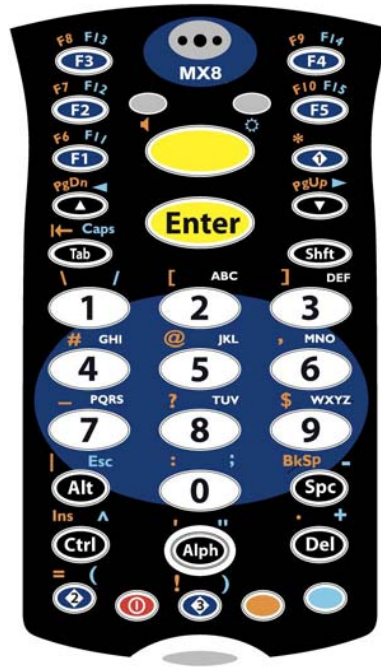
Figure A-8 MX8 Keypad

Note: All RFTerm shortcut keys are directly accessible through the RFTerm user interface.

When running RFTerm on the MX8, please refer to the MX8 Reference Guide for key function information / descriptions.

When using a sequence of keys that includes the Orange or Blue keys, press the color key first then the rest of the key sequence. Ctrl, Alt, Shft, Blue and Orange keys are "sticky keys". They do not need to be held down before pressing the next (or desired) key. It is valid to use combined modifiers on specific keys.

Alphabetic keys default to lower case letters. Press the Shft key, then the alphabetic key for an uppercase letter.

Note: When the computer boots, the default condition of Caps (or CapsLock) is Off. The Caps (or CapsLock) condition can be toggled with Blue plus Tab key sequence.

RFTerm Functions – MX8

RFTerm Function	Press these keys and then…			Press this MX8 key	RFTerm GUI Equivalent
	Orange or Blue	Ctrl	Shft		
Connect to Host	Alpha	x	x	2 2 2	Session	Connect
Disconnect from Host	Alpha	x	x	3	Session	Disconnect
Exit the Emulator	Alpha			ALT + 9 9	File	Exit
Hide / Display All Bars	Alpha	x	x	5 5 5	File	Settings	Display	Hide All Bars
Hide / Display MenuBar	Alpha	x	x	6	File	Settings	Display	Hide MenuBar
Hide / Display ToolBar	Alpha	x	x	8	File	Settings	Display	Hide ToolBar
Hide / Display KeyBar	Alpha	x	x	5 5	File	Settings	Display	Hide KeyBar
Toggle MenuBar Bottom / Top	Alpha	x	x	2 2	File	Settings	Display	Menu at Bottom
Configure Host Session	Alpha	x	x	6 6	Session	Configure
Application Settings	Alpha	x	x	3 3	File	Settings
Increase Font Size		x	x	Up Arrow	Display	Font Up
Decrease Font Size		x	x	Down Arrow	Display	Font Down
Toggle Font Bold / Normal		x	x	1	Session	Configure	Font
Display / Hide SmartPad	Alpha	x	x	7	Display	SmartPad
Display / Hide InputTool	Alpha	x	x	4 4 4	Display	InputTool
Help	Alpha	x		4 4	
Scroll Up One Row			x	Up Arrow	Scroll Bar
Scroll Down One Row			x	Down Arrow	Scroll Bar
Scroll Left One Row	Blue		x	Down Arrow	Scroll Bar
Scroll Right One Row	Blue		x	Up Arrow	Scroll Bar

ANSI Functions – MX8

When running the RFTerm ANSI on the MX8, please refer to the **ANSI Plus Reference Guide** for descriptions/explanations of the ANSI functions:

Note: Physical keypad key presses and MX8 virtual keypad presses are the same.

ANSI Function	Press these keys and then…				Press this MX8 key	RFTerm GUI Equivalent
	Orange or Blue	Shft	Ctrl	Alt		
Answerback					**	Display | SmartPad | Ans
Backspace	Orange				BKSP	Display | SmartPad | BS
Delete					DEL	Display | SmartPad | Del
Exit Program [9]	Alpha			x	9 9	File | Exit
Function Key Editor					**	Session | Configure | FNKeys
Help	Alpha		x		4 4	
Print Screen	Alpha		x	x	7	
Send					ENT	Display | Smartpad | Ent
Window Down			x		Down Arrow	Scrollbar
Window Left	Blue		x		Down Arrow	Scrollbar
Window Right	Blue		x		Up Arrow	Scrollbar
Window Up			x		Up Arrow	Scrollbar
F1 – F5 and F6 – F10	Orange				F1 – F5	Display | SmartPad | F1 – F10
F11 – F15	Blue				F1 – F5	Display | SmartPad | F11 – F20
F16 – F20		x			F1 – F5	
F21 – F24	Orange		x		F1 – F4	KeyBar
F25 – F40					**	KeyBar

** Not Supported

---

[9] The MX7 Virtual Keypad accepts only Shift+Ctrl+X as an Exit Program command.

IBM 3270 Functions – MX8

When running the RFTerm IBM 3270 terminal emulator on the MX8, please refer to the **TN3270 Terminal Emulation Reference Guide** for descriptions/explanations of the 3270 functions.

3270 Function	Press these keys and then…				Press this MX8 key	RFTerm GUI Equivalent
	Orange or Blue	Shft	Ctrl	Alt		
Attention	Alpha		x		2	Display | SmartPad | Attn
Backtab	Orange				Tab	Display | SmartPad | BTb
Clear	Alpha		x		2 2 2	Display | SmartPad | Clr
Delete	Alpha		x		3	Display | SmartPad | Del
Erase	Orange				Spc	Display | SmartPad | Bsp
Erase Input	Orange		x		Spc	Display | SmartPad | Eln
Error Reset or Reset	Alpha		x		7 7 7	Display | SmartPad | Rst
Exit Program	Alpha			x	9	File | Exit
Function Key Editor					**	Session | Configure | FNKeys
Help	Alpha		x		4 4	
Home Key					**	Display | SmartPad | Hm
Insert	Alpha		x		4 4 4	Display | SmartPad | Ins
New Line Key	Alpha		x		6 6	Display | SmartPad | Nl
Next or Tab					Tab	Display | SmartPad | Tb
Print Screen	Alpha		x	x	7	
Status Line Toggle	Alpha			x	8	File | Settings | Display | Hide O1A
SW Rev						File | About
Sys_Req	Alpha		x		7 7 7 7	Display | SmartPad | Srq
Window Down			x		Down Arrow	Scrollbar
Window Right	Blue		x		Up Arrow	Scrollbar
Window Left	Blue		x		Down Arrow	Scrollbar
Window Up			x		Up Arrow	Scrollbar
PA1 – PA3			x		1 – 3	Display | SmartPad | Pa1 – Pa3
F1 – F5					F1 – F5	Display | SmartPad | F1 – F10
F6 – F10	Orange					
F11 – F15	Blue				F1 – F5	Display | SmartPad | F11 – F20
F16 – F20		x				
F21 – F24	Orange		x		F1 – F4	Display | SmartPad | F21 – F24

**   Not Supported

IBM 5250 Functions – MX8

When running the RFTerm IBM 5250 terminal emulator on MX8 devices, please refer to the **TN5250 Terminal Emulation Reference Guide** for descriptions/explanations of the 5250 functions.

5250 Function	Press these keys and then…				Press this MX8 key	RFTerm GUI Equivalent
	Orange or Blue	Shft	Ctrl	Alt		
Attention	Alpha		x		2	Display | SmartPad | Attn
Backtab	Orange				Tab	Display | SmartPad | Btb
Char Backspace	Orange				Spc	Display | SmartPad | BSp
Clear	Alpha		x		2 2 2	Display | SmartPad | Clr
Delete	Alpha		x		3	Display | SmartPad | Del
Dup	Alpha		x		8 8	Display | SmartPad | Dup
Erase Input	Orange		x		Spc	Display | SmartPad | Eln
Error Reset or Reset	Blue			x		Display | SmartPad | Rst
Exit Program	Alpha			x	9 9	File | Exit
Field Exit			x		Enter	Display | SmartPad | FEx
					Diamond 1	
Field Minus	Alpha		x		6	Display | SmartPad | F-
Field Plus	Alpha		x		5 5 5	Display | SmartPad | F+
Function Key Editor					**	Session | Configure | FNKeys
Help	Alpha		x		4 4	
Home Key					**	Display | SmartPad | Hm
Insert	Alpha		x		4 4 4	Display | SmartPad | Ins
New Line Key	Alpha		x		6 6	Display | SmartPad | Nl
Next or Field Advance or Tab					Tab	Base KeyBar: Tb
Print Screen	Alpha		x	x	7	Display | SmartPad | Prt
Status Line Toggle	Alpha			x	8	File | settings | Display | Hide O1A
Sys_Req	Alpha		x		7 7 7 7	Display | SmartPad | Srq
SW Rev					**	File | About
Send or Enter CATV					Enter	Display | SmartPad | Ent
Window Down			x		Down Arrow	Scrollbar
Window Left	Blue		x		Down Arrow	Scrollbar
Window Right	Blue		x		Up Arrow	Scrollbar
Window Up			x		Up Arrow	Scrollbar

5250 Function	Press these keys and then…				Press this MX8 key	RFTerm GUI Equivalent
	Orange or Blue	Shft	Ctrl	Alt		
F1 – F5					F1 – F5	Display | SmartPad | F1 – F10
F6 – F10	Orange					
F11 – F15	Blue				F1 – F5	Display | SmartPad | F11 – F20
F16 – F20		x				
F21 – F24	Orange		x		F1 – F4	Display | SmartPad | F21 – F24

**   Not Supported

VX5 Key Maps

Figure A-9 VX5 95-Key QWERTY Keyboard

Figure A-10 VX5 60-Key QWERTY Keyboard

Note: All RFTerm shortcut keys are directly accessible through the RFTerm user interface.

When running RFTerm on the VX5 (Windows XP/2000), please refer to the specific Windows XP/2000 user guides for NumLock and CapsLock setting, if needed. Please refer to the VX5 Reference Guide for VX5 keypresses.

The 2nd key function is available on the 60-key keyboard *only*. When using a sequence of keys that includes the 2nd key, press the 2nd key first then the rest of the key sequence.

Note: There is no SmartPad capability in Windows XP or Windows 2000 devices.

RFTerm Functions – VX5 (XP / 2000)

RFTerm Function	Press these keys and then…			Press this VX5 key	RFTerm GUI Equivalent
	2nd	CTRL	SHIFT		
Connect to Host		x	x	C	Session | Connect
Disconnect from Host		x	x	D	Session | Disconnect
Configure Host Session		x	x	N	Session | Configure
Hide / Display KeyBar and ToolBar		x	x	L	File | Settings | Display | Hide All Bars
Hide / Display ToolBar		x	x	T	File | Settings | Display | Hide ToolBar
Hide / Display KeyBar		x	x	K	File | Settings | Display | Hide KeyBar
Application Settings		x	x	E	File | Settings
Increase Font Size		x	x	Up Arrow	ToolBar: A with Up Arrow
Decrease Font Size		x	x	Down Arrow	ToolBar: A with Down Arrow
Toggle Font Bold / Normal		x	x	1	Session | Configure | Font | Font combo box
Scroll Up One Row			x	Up Arrow	Scrollbar or Scroll KeyBar
Scroll Down One Row			x	Down Arrow	Scrollbar or Scroll KeyBar
Scroll Left One Row			x	Left Arrow	Scrollbar or Scroll KeyBar
Scroll Right One Row			x	Right Arrow	Scrollbar or Scroll KeyBar
Exit RFTerm		x	x	X	File | Exit

ANSI Functions – VX5 (XP / 2000)

When running the RFTerm ANSI on LXE Windows XP or Windows 2000 devices, please refer to the **ANSI Plus Reference Guide** for descriptions/explanations of the ANSI functions.

Note: There is no SmartPad capability in Windows XP or Windows 2000 devices.

ANSI Function	Press these keys and then…				Press this VX5 key	RFTerm GUI Equivalent
	2nd	SHIFT	CTRL	ALT		
Answerback						VT 2 KeyBar: Ans
Backspace					BkSp	VT 2 KeyBar: BS
Delete	x [10]				. (period)	VT 2 KeyBar: DEL
Exit Program				x	X	File	Exit
		x	x		X	
F1 – F10					F1 – F10	F1-F8, F9-F16 KeyBars
F11 – F20				x	F1 – F10	F9-F16, F17-F24 KeyBars
F21 – F30		x			F1 – F10	F17-F24, F25-F32 KeyBars
F31 – F40			x		F1 – F10	F25-F32, F33-F40 KeyBars
Local Echo On / Off						Session	Configure	Connection	VT Config	Screen	Local Echo
Print Screen			x	x	P	
Send					Enter	Base KeyBar: Ent
Window Down			x		Down Arrow	Scrollbar or Scroll KeyBar
Window Left			x		Left Arrow	Scrollbar or Scroll KeyBar
Window Right			x		Right Arrow	Scrollbar or Scroll KeyBar
Window Up			x		Up Arrow	Scrollbar or Scroll KeyBar

IBM 3270 Functions – VX5 (XP / 2000)

When running the RFTerm IBM 3270 terminal emulator on LXE Windows XP or Windows 2000 devices, please refer to the **TN3270 Terminal Reference Guide** for descriptions/explanations of the 3270 functions.

Note: There is no SmartPad capability in Windows XP or Windows 2000 devices.

3270 Function	Press these keys and then…				Press this VX5 key	RFTerm GUI Equivalent
	2nd	SHIFT	CTRL	ALT		
Attention			x		A	IBM 1 KeyBar: Atn
Backtab	x [5]				Tab	Base KeyBar: Btb
Clear			x		C	IBM 1 KeyBar: Clr
Cursor Block / Underline						Session	Configure	Cursor	Cursor Options
Delete			x		D	IBM 2 KeyBar: Dup
Erase Input			x		BkSp	IBM 1 KeyBar: Er INP

---

[10] 2nd key is available on the 60-key keyboard only.

3270 Function	Press these keys and then…				Press this VX5 key	RFTerm GUI Equivalent
	2nd	SHIFT	CTRL	ALT		
Erase					BkSp	
Error Reset			x		R	IBM 1 KeyBar: Rst
Exit Program				x	X	File | Exit
		x	x		X	
Home Key A Home Key B			x		Left Arrow	IBM 2 KeyBar: Home
Insert			x		I	IBM 2 KeyBar: Ins
New Line Key			x		N	IBM 2 KeyBar: NL
Next or Tab					Tab	Base KeyBar: Tb
PA1 – PA3			x		F1 – F3	IBM 1 KeyBar: PA1 – PA3
PF1 – PF10					F1 – F10	F1-F8, F9-F16 KeyBars
PF11 – PF20				x	F1 – F10	F9-F16, F17-F24 KeyBars
PF21 – PF24		x			F1 – F4	F25-F32 KeyBar
Print Screen			x	x	P	
Roll Down		x			Down Arrow	IBM 3 KeyBar: Roll Dn
Roll Up		x			Up Arrow	IBM 3 KeyBar: Roll Up
Send					Enter	Base KeyBar: Ent
Status Line Toggle				x	T	Special KeyBar: OIA
		x	x		O	
SW Rev				x	F then A	File | About
System Request			x		S	IBM 2 KeyBar: Sys Rq
Window Down			x		Down Arrow	Scrollbar or Scroll KeyBar
Window Left			x		Left Arrow	Scrollbar or Scroll KeyBar
Window Right			x		Right Arrow	Scrollbar or Scroll KeyBar
Window Up			x		Up Arrow	Scrollbar or Scroll KeyBar

IBM 5250 Functions – VX5 (XP / 2000)

When running the RFTerm IBM 5250 terminal emulator on LXE Windows XP or Windows 2000devices, please refer to the **TN5250 Terminal Reference Guide** for descriptions/explanations of the 5250 functions.

Note: *There is no SmartPad capability in Windows XP or Windows 2000 devices.*

5250 Function	Press these keys and then…				Press this VX5 key	RFTerm GUI Equivalent
	2nd	SHIFT	CTRL	ALT		
Attention			x		A	IBM 1 KeyBar: Atn
Backtab	x [11]				TAB	Base KeyBar: Btb
Char Backspace					BkSp	
Clear			x		C	IBM 1 KeyBar: Clr
Cursor Block / Underline						File | Settings | Cursor | Cursor Options

---

[11] 2nd key is available on the 60-key keyboard only.

5250 Function	Press these keys and then…				Press this VX5 key	RFTerm GUI Equivalent
	2nd	SHIFT	CTRL	ALT		
Delete			x		D	IBM 2 KeyBar: DEL
Dup			x		U	IBM 2 KeyBar: Dup
Erase Input			x		BkSp	IBM 1 KeyBar: Er INP
Error Reset or Reset					ESC	Base KeyBar: ESC
Exit Program				x	X	File | Exit
		x	x		X	
F1 – F10					F1 – F10	F1-F8, F9-F16 KeyBars
F11 – F20				x	F1 – F10	F9-F16, F17-F24 KeyBars
F21 – F24		x			F1 – F4	F25-F32 KeyBar
Field Exit			x		Enter	IBM 2 KeyBar: Fld EXIT
	x [6]				End	
					Pause [12]	
Field Minus			x		M	IBM 3 KeyBar: Fld-
Field Plus			x		L	IBM 3 KeyBar: Fld+
Home Key					Home	IBM 2 KeyBar: Home
Insert			x		I	IBM 2 KeyBar: Ins
New Line Key			x		N	IBM 2 KeyBar: NL
Next or Field Advance or Tab					Tab	Base KeyBar: Tb
Print Screen			x	x	P	IBM 3 KeyBar: Prnt
Roll Down		x			Down Arrow	IBM 3 KeyBar: Roll Dn
Roll Up		x			Up Arrow	IBM 3 KeyBar: Roll Up
Send or Enter CATV					Enter	Base KeyBar: Ent
Status Line Toggle				x	T	Special KeyBar: OIA
		x	x		O	
SW Rev				x	F then A	File | About
System Request			x		S	IBM 2 KeyBar: Sys Rq
Window Down			x		Down Arrow	Scrollbar or Scroll KeyBar
Window Left			x		Left Arrow	Scrollbar or Scroll KeyBar
Window Right			x		Right Arrow	Scrollbar or Scroll KeyBar
Window Up			x		Up Arrow	Scrollbar or Scroll KeyBar

[12] Available on the 96-key keyboard only.

VX6 and VX7 Key Maps

Note: All RFTerm shortcut keys are directly accessible through the RFTerm user interface.

When running RFTerm on the VX6 or VX7, please refer to the specific user guides for inclusive information.

Unless otherwise noted, each keypress sequence is valid for:

VX6 with integrated keypad

Figure A-11 VX6 QWERTY Keyboard

VX7 with 60-key Keyboard

Figure A-12 VX7 60-key QWERTY Keyboard

For the VX6 and the VX7 with 60-key keyboard:

- When using a sequence of keys that includes the 2nd key, press the 2nd key first then the rest of the key sequence.

- Set the On/Off condition of NumLock before pressing a key sequence. There is no visual indication of the condition of NumLock.

- When the computer boots, the default condition of Caps (or CapsLock) is Off. The Caps (or CapsLock) condition can be toggled with a 2nd-F1 key sequence. The CAPS LED is illuminated when CapsLock is On.

- When the computer boots, the default condition of NumLock is On. The warmboot behavior of NumLock can be configured, as detailed in the VX6 Reference Guide or the VX7 Reference Guide.

VX7 with 95-key keyboard

Figure A-13 VX7 95-key QWERTY Keyboard

For the VX7 with 95-key keyboard:

- There is no 2nd key on this keyboard.

- Set the On/Off condition of NumLock before pressing a key sequence. When NumLock is On, the NumLock key and the numeric keys are backlit **amber**. When NumLock is Off, these keys are backlit **green**.

- When the computer boots, the default condition of CapsLock is Off. The Caps condition can be toggled with a CapsLock key. The CapsLock key is backlit **amber** when CapsLock is On and backlit **green** when CapsLock is Off.

- When the computer boots, the default condition of NumLock is On. The warmboot behavior of NumLock can be configured, as detailed in the VX7 Reference Guide.

RFTerm Functions – VX6 and VX7 (CE .NET)

RFTerm Function	Press these keys and then…				Press this VX6 / VX7 key	RFTerm GUI Equivalent
	2nd	CTRL	SHIFT	ALT		
Application Settings		x	x		E	File | Settings
Configure Host Session		x	x		N	Session | Configure
Connect to Host		x	x		C	Session | Connect
Decrease Font Size		x	x		Down Arrow	Display | Font Down
Disconnect from Host		x	x		D	Session | Disconnect
Display / Hide Smart Pad		x	x		P	Display | Smart Pap
Display / Hide Text Input		x	x		I	Display | Input Tool
Exit RFTerm		x	x		X	File | Exit
				x	X	
Help		x			H	
Hide / Display All Bars		x	x		L	File | Settings | Display | Hide All Bars
Hide / Display Menu Bar		x	x		M	File | Settings | Display | Hide MenuBar

RFTerm Function	Press these keys and then…				Press this VX6 / VX7 key	RFTerm GUI Equivalent
	2nd	CTRL	SHIFT	ALT		
Hide / Display Key Bar		x	x		K	File | Settings | Display | Hide KeyBar
Hide / Display ToolBar		x	x		T	File | Settings | Display | Hide ToolBar
Increase Font Size		x	x		Up Arrow	Display | Font Up
Scroll Down One Row			x		Down Arrow	Scrollbar
Scroll Left One Row			x		Left Arrow	Scrollbar
Scroll Right One Row			x		Right Arrow	Scrollbar
Scroll Up One Row			x		Up Arrow	Scrollbar
Toggle All Bars Bottom / Top		x	x		B	File | Settings | Display | At Bottom
Toggle Font Bold / Normal		x	x		1	Session | Configure | Font Tab

ANSI Functions – VX6 and VX7 (CE .NET)

When running RFTerm ANSI on LXE Windows CE .NET devices, please refer to the **ANSI Plus Reference Guide** for descriptions and explanations of the ANSI functions.

ANSI Function	Press these keys and then…				Press this VX6 / VX7 key	RFTerm GUI Equivalent
	2nd	SHIFT	CTRL	ALT		
Answerback						Display | Smart Pad | Ans
Backspace					BkSp	Display | Smart Pad | BS
Delete	x				Del [13]	Display | Smart Pad | Del
					Del [14]	
Exit Program				x	X	File | Exit
		x	x		X	
Help			x		H	
F1 – F10					F1 – F10	Display | Smart Pad | F6 – F10
F11 – F20				x	F1 – F10	Display | Smart Pad | F11 – F20
F21 – F30		x			F1 – F10	
F31 – F40			x		F1 – F10	
Function Key Editor						Session | Configure | FNKeys
Local Echo						Session | Configure | Connection | VT Config | Screen | Local Echo
Print Screen			x	x	P	
Send					Enter	Display | Smart Pad | Ent

---

[13] For VX6 keypad and VX7 60-key keyboard only.
[14] For VX7 with 95-key keyboard only.

ANSI Function	Press these keys and then…				Press this VX6 / VX7 key	RFTerm GUI Equivalent
	2nd	SHIFT	CTRL	ALT		
Window Down			x		Down Arrow	Scrollbar
Window Left			x		Left Arrow	Scrollbar
Window Right			x		Right Arrow	Scrollbar
Window Up			x		Up Arrow	Scrollbar

IBM 3270 Functions – VX6 and VX7 (CE .NET)

When running RFTerm IBM 3270 on LXE Windows CE .NET devices, please refer to the **TN3270 Terminal Reference Guide** for descriptions and explanations of the 3270 functions.

3270 Function	Press these keys and then…				Press this VX6 / VX7 key	RFTerm GUI Equivalent
	2nd	SHIFT	CTRL	ALT		
Attention			x		A	Display | Smart Pad | Attn
Backtab	x				Tab [15]	Display | Smart Pad | BTb
		x			Tab [16]	
Clear			x		C	Display | Smart Pad | Clr
					Esc	
Cursor Block / Underline						File | Settings | Cursor | Cursor Options
Delete			x		D	Display | Smart Pad | Del
Erase					BkSp	Display | Smart Pad | BSp
Erase Input			x		BkSp	Display | Smart Pad | Eln
Error Reset			x		R	Display | Smart Pad | Rst
Exit Program				x	X	File | Exit
		x	x		X	
Function Key Editor						Session | Configure | FNKeys
Help			x		H	
Home Key			x		Left Arrow	Display | Smart Pad | Hm
	x				Home [10]	
			x		Home [11]	
Insert			x		I	Display | Smart Pad | Ins
LDUB (Not Supported)						
New Line Key			x		N	Display | Smart Pad | Nl
Next or Tab					Tab	Display | Smart Pad | Tb
PA1 – PA3			x		F1 – F3	Display | Smart Pad | PA1 – PA3
PF1 – PF10					F1 – F10	Display | Smart Pad | F1 – F10

---

[15] For VX6 keypad and VX7 60-key keyboard only.
[16] For VX7 60-key keyboard only.

3270 Function	Press these keys and then…				Press this VX6 / VX7 key	RFTerm GUI Equivalent
	2nd	SHIFT	CTRL	ALT		
PF11 – PF20				x	F1 – F10	Display | Smart Pad | F11 – F20
PF21 – PF24		x			F1 – F4	Display | Smart Pad | F21 – F24
Print Screen			x	x	P	
RDUB (Not Supported)						
Status Line Toggle				x	T	File | Settings | Display | Hide OIA
		x	x		O	
System Request			x		S	Display | Smart Pad | Srq
SW REV				x	F then A	File | About
Window Down			x		Down Arrow	Scrollbar
Window Left			x		Left Arrow	Scrollbar
Window Right			x		Right Arrow	Scrollbar
Window Up			x		Up Arrow	Scrollbar

IBM 5250 Functions – VX6 and VX7 (CE .NET)

When running RFTerm IBM 5250 on LXE Windows CE .NET devices, please refer to the **TN5250 Terminal Reference Guide** for descriptions and explanations of the 5250 functions.

5250 Function	Press these keys and then…				Press this VX6 / VX7 key	RFTerm GUI Equivalent
	2nd	SHIFT	CTRL	ALT		
Attention			x		A	Display | Smart Pad | Attn
Backtab	x				Tab [17]	Display | Smart Pad | Btb
		x			Tab [18]	
Char Backspace					BkSp	Display | Smart Pad | BSp
Clear			x		C	Display | Smart Pad | Clr
Cursor Block / Underline						File | Settings | Cursor | Cursor Options
Delete			x		D	Display | Smart Pad | Del
Dup			x		U	Display | Smart Pad | Dup
Erase Input			x		BkSp	Display | Smart Pad | Eln
Error Reset or Reset					Esc	Display | Smart Pad | Rst
Exit Program				x	X	File | Exit
		x	x		X	
Fast Backward (Not Supported)						
Fast Forward (Not Supported)						

---

[17] For VX6 keypad and VX7 60-key keyboard only.
[18] For VX7 60-key keyboard only.

5250 Function	Press these keys and then…				Press this VX6 / VX7 key	RFTerm GUI Equivalent
	2nd	SHIFT	CTRL	ALT		
F1 – F10					F1 – F10	Display | Smart Pad | F1 – F10
F11 – F20				x	F1 – F10	Display | Smart Pad | F11 – F20
F21 – F24		x			F1 – F4	Display | Smart Pad | F21 – F24
Field Exit			x		Enter	Display | Smart Pad | FEx
	x				End[19]	
					End[20]	
					Pause [16]	
Field Minus			x		M	Display | Smart Pad | F-
Field Plus			x		L	Display | Smart Pad | F+
Function Key Editor						Session | Configure |FNKeys
Help			x		H	
Home Key	x				Home [15]	Display | Smart Pad | Hm
					Home [16]	
Insert			x		I	Display | Smart Pad | Ins
New Line Key			x		N	Display | Smart Pad | Nl
Next or Field Advance or Tab					Tab	Display | Smart Pad | Tb
Print Screen			x	x	P	Display | Smart Pad | Prt
Send or Enter CATV					Enter	Display | Smart Pad | Ent
Status Line Toggle				x	T	File | Settings | Display | Hide OIA
		x	x		O	
Sys Req			x		S	Display | Smart Pad |Srq
SW Rev				x	F then A	File | About
Window Down			x		Down Arrow	Scrollbar
Window Left			x		Left Arrow	Scrollbar
Window Right			x		Right Arrow	Scrollbar
Window Up			x		Up Arrow	Scrollbar

---

[19] For VX6 keypad and VX7 60-key keyboard only.
[20] For VX7 60-key keyboard only.

Appendix B ANSI Quick Reference Guide

Functionality List

Notes

Reset mode terminating character is l, set mode is h.

C	The configuration utility equivalent of this command.
M	The modes that affect this command.
R	The effect generated when this mode is reset.
S	The effect generated when this mode is set.

BKC	Bar code/Keypad Setup Private Sequence
CCT	Character Collection Timer Private Sequence
DECOM	DEC Origin Mode
DECPEX	DEC Print Extent Mode
DECPFF	DEC Print Form Feed Mode
DRM/SM	DECRM/DECSM
EBM	Editing Boundary Mode
ERM	Erasure Mode
LNM	Line Feed/New Line Mode
LRM/SM	LXERM/LXESM
NRC	National Replacement Character Sets
SEM	Select Editing Extent Mode
SGR	Select Graphic Rendition
SRM	Send/Receive Mode
TSM	Tab Stop Mode
TSS	Transmit/Screen Setup Private Sequence

Cursor Positioning

Cursor Pos.	Description	Hex Value / Esc Seq	Notes
BS	Backspace	0x08 (^H)	
CUB	Cursor Back	ESC[PnD	M=None
CUF	Cursor Forward	ESC[PnC	M=None
CUD	Cursor Down	ESC[PnB	M=None
CUU	Cursor Up	ESC[PnA	M=None
CUP	Cursor Position	ESC[Pr;PcH	M=None
HVP	Horizontal Vertical Pos	ESC[Pr;Pcf	M=None
CHA	Cursor Horizontal Absolute	ESC[PnG	
CR	Carriage Return	0x0d (^M)	
LF	Line Feed	0x0a (^J)	M=Scroll Region, LNM
FF	Form Feed	0x0c (^L)	M=Scroll Region, LNM; Same as LF
VT	Vertical Tab	0x0b (^K)	M=Scroll Region, LNM; Same as LF
IND	Index	0x84 (ESC D)	M=Scroll Region
RI	Reverse Index	0x8d (ESC M)	M=Scroll Region
LNM	Line Feed/New Line Mode	ESC[20l (or h)	R=LF, S=LF+CR (RM/SM); See Transmission to Host
LXENE	New Line Echo Mode	ESC[=2l (or h)	R=neither, S=CR+LF (LRM/SM); Also TSS 6; C=EchoCrLf
CNL	Cursor Next Line	ESC[PnE	
NEL	Next Line	0x85 (ESC E)	M=Scroll Region
CVT	Cursor Vertical Tab	ESC[PnY	
CPL	Cursor Previous Line	ESC[PnF	

Forms

Forms	Description	Hex Value/ ESC Seq	Notes
HT	Horizontal Tab	0x09 (^I)	No tab to protected cells
CHT	Cursor Horizontal Tab	ESC[PnI	No tab to protected cells
CBT	Cursor Back Tab	ESC[PnZ	No tab to protected cells
HTJ	Horizontal Tab w/Justify	0x89 (ESC I)	No tab to protected cells
HTS	Horizontal Tab Set	0x88 (ESC H)	M=TSM
CTC	Cursor Tab Clear	ESC[PnW	(TSM reset/TSM set); 0,2=column/position, 4=all/row, 5=all/all

Forms	Description	Hex Value/ ESC Seq	Notes
TBC	Tab Clear	ESC[Png	(TSM reset/TSM set); 0=column/position, 2=all/row, 3=all/all
TSM	Tab Set Mode	ESC[18l (or h)	R=virtual, S=curr line (RM/SM)
DAQ	Define Area Qualification	ESC[Pno	0=all data, 1=prot, 2=display, 3=num, 4=alpha, 5=right just, 6=zero filled, 7=tab, 8=selected, 9=space filled, 10=hidden, 15=keyboard input only
SPA	Start Protected Area	0x96 (ESC V)	Same as DAQ 1
EPA	End Protected Area	0x97 (ESC W)	Current area is all input
SSA	Start Selected Area	0x86 (ESC F)	
ESA	End Selected Area	0x87 (ESC G)	Current area is all input
EA	Erase Area	ESC[PnO	0=to end, 1=from start, 2=all; M=ERM Does not erase area type
EF	Erase Field	ESC[PnN	0=to end, 1= from start, 2=all; M=EBM; Does not erase area type
LEM	Local Edit Match	ESC[Pn+A	R=display, S=hide
LEM Alpha WC	Alpha only wildcard	ESC[c1;;p2+G	Matches A-Z, a-z, comma, period, dash, space; C=AlphaOnly
LEM Digits WC	Digits only wildcard	ESC[c1;;;p3+G	Matches 0-9; C=DigitsOnly
LEM Numeric WC	Numeric only wildcard	ESC[c1;;;;p4+G	Matches 0-9, plus, comma, period, dash, space; C=NumericOnly
LEM Alphanum WC	Alphanum only wildcard	ESC[c1;;;;;p5+G	Matches A-Z, a-z, 0-9, plus, comma, period, dash, space; C = AlphaNumericOnly
LEM Match All WC	Match all wildcard	ESC[c1;;;;;;p6+G	All GL and GR; C=MatchAll

Transmission To Host

Transmission To Host	Description	Hex Value/ Esc Seq	Notes
GATM	Guarded Area Transmit Mode	ESC[1l (or h)	R=unguarded, S=entire screen (RM/SM)
MATM	Multiple Area Transmit Mode	ESC[15l (or h)	R=cursor selected, S=all selected (RM/SM)
SATM	Selected Area Transmit Mode	ESC[17l (or h)	R=selected, S=virtual (RM/SM)
LXEAS	Auto Send Mode	ESC[=3l (or h)	R=disable, s=enable (LRM/SM); SRM must be reset; Also TSS 3; C=AutoTransmit
SRM	Send Receive Mode	ESC[12l (or h)	R=local echo, S=char (RM/SM); Also TSS 10; C=SendReceiveMode
LXELM	Line Mode	ESC[=1l (or h)	R=block, S=scroll (LRM/SM); Also TSS 5; C=ScrollMode
DECSCL	Select Operating Level	ESC[62;Pn"p	R=8 bit, S=7 bit
S7C1T	Select 7 Bit Transmission	ESC F	A space (0x20) precedes F; C=EightBitMode
S8C1T	Select 8 Bit Transmission	ESC G	A space (0x20) precedes G; C=EightBitMode
LNM	Line Feed/New Line Mode	ESC[c1;;;;p4+F	R=CR, S= CR+LF; Also TSS 4; See cursor positioning; C=TransmitCR/LF
OLM	Online Mode	ESC[c1;;;;;;;;p8+F	R=online, S=offline; Also TSS 8; C=OnlineMode

Editing

Editing	Description	Hex Value/ Esc Seq	Notes
ECH	Erase Character	ESC[PnX	M=ERM
DCH	Delete Character	ESC[PnP	M=SEM Aborts at protected cells
ICH	Insert Character	ESC[Pn@	M=SEM Aborts at protected cells
SEM	Select Editing Extent Mode	ESC[PnQ	0=display, 1=row, 2=field, 3=area; M=None
DL	Delete Line	ESC[PnM	M=None
IL	Insert Line	ESC[PnL	M=None
ED	Erase Display	ESC[PnJ	0=to end, 1=from start, 2=all; M=ERM Erases area type/boundaries, homes cursor.
EL	Erase Line	ESC[PnK	0=to end, 1=from start, 2=all; M=ERM Erases area type/boundaries
DECSED	DEC Selective Erase In Display	ESC[?PnJ	0=to end, 1=from start, 2=all; M=None Never erases protected cells or area types/boundaries, homes cursor
DECSEL	DEC Selective Erase In Line	ESC[?PnK	0=to end, 1=from start, 2=all; M=None Never erases protected cells or area types/boundaries
ERM	Erasure Mode	ESC[6l (or h)	R=cannot erase prot, S=erase prot (RM/SM)
SD	Scroll Down	ESC[PnT	M=Scroll Region
SU	Scroll Up	ESC[PnS	M=Scroll Region
DECSTBM	DEC Set Top/ Bottom Margins	ESC[Pt;Pbr	M=DECOM; Must have full screen client to use; Min height of region is 2 rows, Homes cursor according to DECOM
IRM	Insert/Replace Mode	ESC[4l (or h)	R=overwrite, S=insert (RM/SM)

Character Rendition

Character Rendition	Description	Hex Value/ Esc Seq	Notes
SGR	Select Graphics Rendition	ESC[Pnm	0=normal, 1,22=bold, 4,24=underline, 5,25=blink, 7,27=reverse; First num enables, second disables

Bar Code

Bar code	Description	Hex Value/ Esc Seq	Notes
Left Justify	Left Justify	ESC[c1;;p2+E	R=off, S=on (BKC); Requires block mode; C=LeftJustify
Clear	Clear	ESC[c1;;;p3+E	R=off, S=on (BKC); Requires block mode; C=ClearData
Truncate	Truncate	ESC[c1;;;;p4+E	R=off, S=on (BKC); Requires block mode; C=TruncateData
Auto Advance	Auto Advance	ESC[c1;;;;;p5+E	R=off, S=on (BKC); Requires block mode; C=AutoAdvance

Reports / Status / Test

Reports /Status/ Test	Description	Hex Value/ Esc Seq	Notes
ENQ	Enquire	0x05 (^E)	
CPR	Cursor Position Report	ESC[Pr;PcR	Only sent from client; Position relative to virtual display
DSR	Device Status Report	ESC[Pnn	5=status, 6=CPR, 26=keyboard

Printing

Printing	Description	Hex Value/ Esc Seq	Notes
MC	Media Copy	ESC[Pni	4=off 5=on
DC2	Device Control 2	0x12 (^R)	Ignored from host
DC4	Device Control 4	0x14 (^T)	Same as MC4
APM	Auto Print Mode	ESC[?Pni	4=off 5=on
PL	Print Line	ECS[?1i	
PS	Print Screen	ESC[0i	
DECPFF	DEC Print Form Feed Mode	ECS[?18l (or h)	h=enable terminating character l=disable terminating character
DECPEX	DEC Print Extent Mode	ECS[?18l (or h)	h=full screen l=scrolling region

Hat Encoding

Desired ASCII	Hex Value	Hat Encoded	Desired ASCII	Hex Value	Hat Encoded
NUL	00	^@	ESA	87	~^G
SOH	01	^A	HTS	88	~^H
STX	02	^B	HTJ	89	~^I
ETX	03	^C	VTS	8A	~^J
EOT	04	^D	PLD	8B	~^K
ENQ	05	^E	PLU	8C	~^L
ACK	06	^F	RI	8D	~^M
BEL	07	^G	SS2	8E	~^N
BS	08	^H	SS3	8F	~^O
HT	09	^I	DCS	90	~^P
LF	0A	^J	PU1	91	~^Q
VT	0B	^K	PU2	92	~^R
FF	0C	^L	STS	93	~^S
CR	0D	^M	CCH	94	~^T
SO	0E	^N	MW	95	~^U
SI	0F	^O	SPA	96	~^V
DLE	10	^P	EPA	97	~^W
DC1 (XON)	11	^Q		98	~^X
DC2	12	^R		99	~^Y
DC3 (XOFF)	13	^S		9A	~^Z
DC4	14	^T	CSI	9B	~^[
NAK	15	^U	ST	9C	~^\\
SYN	16	^V	OSC	9D	~^]
ETB	17	^W	PM	9E	~^^
CAN	18	^X	APC	9F	~^_ (Underscore)
EM	19	^Y	(no-break space)	A0	~ (Tilde and Space)
SUB	1A	^Z	¡	A1	~!
ESC	1B	^[	¢	A2	~"
FS	1C	^\\	£	A3	~#
GS	1D	^]	¤	A4	~$
RS	1E	^^	¥	A5	~%
US	1F	^_ (Underscore)	¦	A6	~&
	80	~^@	§	A7	~'
	81	~^A	¨	A8	~(
	82	~^B	©	A9	~)
	83	~^C	ª	AA	~*
IND	84	~^D	«	AB	~+
NEL	85	~^E	¬	AC	~,
SSA	86	~^F	(soft hyphen)	AD	~- (Dash)

Hat Encoded Characters Hex 00 through AD

Desired ASCII	Hex Value	Hat Encoded	Desired ASCII	Hex Value	Hat Encoded
®	AE	~. (Period)	×	D7	~W
¯	AF	~/	Ø	D8	~X
°	B0	~0 (Zero)	Ù	D9	~Y
±	B1	~1	Ú	DA	~Z
²	B2	~2	Û	DB	~[
³	B3	~3	Ü	DC	~\\
´	B4	~4	Ý	DD	~]
µ	B5	~5	Þ	DE	~\^
¶	B6	~6	ß	DF	~_ (Underscore)
·	B7	~7	à	E0	~`
¸	B8	~8	á	E1	~a
¹	B9	~9	â	E2	~b
º	BA	~:	ã	E3	~c
»	BB	~;	ä	E4	~d
¼	BC	~<	å	E5	~e
½	BD	~=	æ	E6	~f
¾	BE	~>	ç	E7	~g
¿	BF	~?	è	E8	~h
À	C0	~@	é	E9	~i
Á	C1	~A	ê	EA	~j
Â	C2	~B	ë	EB	~k
Ã	C3	~C	ì	EC	~l
Ä	C4	~D	í	ED	~m
Å	C5	~E	î	EE	~n
Æ	C6	~F	ï	EF	~o
Ç	C7	~G	ð	F0	~p
È	C8	~H	ñ	F1	~q
É	C9	~I	ò	F2	~r
Ê	CA	~J	ó	F3	~s
Ë	CB	~K	ô	F4	~t
Ì	CC	~L	õ	F5	~u
Í	CD	~M	ö	F6	~v
Î	CE	~N	÷	F7	~w
Ï	CF	~O	ø	F8	~x
Ð	D0	~P	ù	F9	~y
Ñ	D1	~Q	ú	FA	~z
Ò	D2	~R	û	FB	~{
Ó	D3	~S	ü	FC	~|
Ô	D4	~T	ý	FD	~}
Õ	D5	~U	þ	FE	~~
Ö	D6	~V	ÿ	FF	~^?

Hat Encoded Characters Hex AE through FF

DEC Special Graphics - Code Table

Refer to the 8-bit code table below that lists all the decimal, hexadecimal, and octal values for each character:

ROW	b4 b3 b2 b1	0 (0 0 0)	1 (0 0 1)	2 (0 1 0)	3 (0 1 1)	4 (1 0 0)	5 (1 0 1)	6 (1 1 0)	7 (1 1 1)
0	0 0 0 0	NUL 0/0/0	DLE 20/16/10	SP 40/32/20	0 60/48/30	@ 100/64/40	P 120/80/50	◆ 140/96/60	— SCAN3 160/112/70
1	0 0 0 1	SOH 1/1/1	DC1 (XON) 21/17/11	! 41/33/21	1 61/49/31	A 101/65/41	Q 121/81/51	▒ 141/97/61	— SCAN5 161/113/71
2	0 0 1 0	STX 2/2/2	DC2 22/18/12	" 42/34/22	2 62/50/32	B 102/66/42	R 122/82/52	H_T 142/98/62	— SCAN7 162/114/72
3	0 0 1 1	ETX 3/3/3	DC3 (XOFF) 23/19/13	# 43/35/23	3 63/51/33	C 103/67/43	S 123/83/53	F_F 143/99/63	— SCAN9 163/115/73
4	0 1 0 0	EOT 4/4/4	DC4 24/20/14	$ 44/36/24	4 64/52/34	D 104/68/44	T 124/84/54	C_R 144/100/64	├ 164/116/74
5	0 1 0 1	ENQ 5/5/5	NAK 25/21/15	% 45/37/25	5 65/53/35	E 105/69/45	U 125/85/55	L_F 145/101/65	┤ 165/117/75
6	0 1 1 0	ACK 6/6/6	SYN 26/22/16	& 46/38/26	6 66/54/36	F 106/70/46	V 126/86/56	° 146/102/66	┴ 166/118/76
7	0 1 1 1	BEL 7/7/7	ETB 27/23/17	' 47/39/27	7 67/55/37	G 107/71/47	W 127/87/57	± 147/103/67	┬ 167/119/77
8	1 0 0 0	BS 10/8/8	CAN 30/24/18	(50/40/28	8 70/56/38	H 110/72/48	X 130/88/58	N_L 150/104/68	│ 170/120/78
9	1 0 0 1	HT 11/9/9	EM 31/25/19	) 51/41/29	9 71/57/39	I 111/73/49	Y 131/89/59	V_T 151/105/69	≤ 171/121/79
10	1 0 1 0	LF 12/10/A	SUB 32/26/1A	* 52/42/2A	: 72/58/3A	J 112/74/4A	Z 132/90/5A	⌟ 152/106/6A	≥ 172/122/7A
11	1 0 1 1	VT 13/11/B	ESC 33/27/1B	+ 53/43/2B	; 73/59/3B	K 113/75/4B	[133/91/5B	┐ 153/107/6B	π 173/123/7B
12	1 1 0 0	FF 14/12/C	FS 34/28/1C	, 54/44/2C	< 74/60/3C	L 114/76/4C	\ 134/92/5C	┌ 154/108/6C	≠ 174/124/7C
13	1 1 0 1	CR 15/13/D	GS 35/29/1D	- 55/45/2D	= 75/61/3D	M 115/77/4D	] 135/93/5D	└ 155/109/6D	£ 175/125/7D
14	1 1 1 0	SO 16/14/E	RS 36/30/1E	. 56/46/2E	> 76/62/3E	N 116/78/4E	^ 136/94/5E	┼ 156/110/6E	· 176/126/7E
15	1 1 1 1	SI 17/15/F	US 37/31/1F	/ 57/47/2F	? 77/63/3F	O 117/79/4F	(BLANK) 137/95/5F	─ SCAN1 157/111/6F	177/127/7F

◄— C0 Codes —► ◄— GL Codes (DEC Special Graphic) —►

KEY

Character — ESC | 23 (Octal) / 27 (Decimal) / 1B (Hexadecimal)

DEC Special Graphics - Code Table

IS0 8859-1 Character Set

	00	01	02	03	04	05	06	07	08	09	0A	0B	0C	0D	0E	0F
00	NUL 0000	STX 0001	SOT 0002	ETX 0003	EOT 0004	ENQ 0005	ACK 0006	BEL 0007	BS 0008	HT 0009	LF 000A	VT 000B	FF 000C	CR 000D	SO 000E	SI 000F
10	DLE 0010	DC1 0011	DC2 0012	DC3 0013	DC4 0014	NAK 0015	SYN 0016	ETB 0017	CAN 0018	EM 0019	SUB 001A	ESC 001B	FS 001C	GS 001D	RS 001E	US 001F
20	SP 0020	! 0021	" 0022	# 0023	$ 0024	% 0025	& 0026	' 0027	(0028	) 0029	* 002A	+ 002B	, 002C	- 002D	. 002E	/ 002F
30	0 0030	1 0031	2 0032	3 0033	4 0034	5 0035	6 0036	7 0037	8 0038	9 0039	: 003A	; 003B	< 003C	= 003D	> 003E	? 003F
40	@ 0040	A 0041	B 0042	C 0043	D 0044	E 0045	F 0046	G 0047	H 0048	I 0049	J 004A	K 004B	L 004C	M 004D	N 004E	O 004F
50	P 0050	Q 0051	R 0052	S 0053	T 0054	U 0055	V 0056	W 0057	X 0058	Y 0059	Z 005A	[005B	\ 005C	] 005D	^ 005E	_ 005F
60	` 0060	a 0061	b 0062	c 0063	d 0064	e 0065	f 0066	g 0067	h 0068	i 0069	j 006A	k 006B	l 006C	m 006D	n 006E	o 006F
70	p 0070	q 0071	r 0072	s 0073	t 0074	u 0075	v 0076	w 0077	x 0078	y 0079	z 007A	{ 007B	| 007C	} 007D	~ 007E	DEL 007F
80																
90																
A0	NBSP 00A0	¡ 00A1	¢ 00A2	£ 00A3	¤ 00A4	¥ 00A5	¦ 00A6	§ 00A7	¨ 00A8	© 00A9	ª 00AA	« 00AB	¬ 00AC	00AD	® 00AE	¯ 00AF
B0	° 00B0	± 00B1	² 00B2	³ 00B3	´ 00B4	µ 00B5	¶ 00B6	· 00B7	¸ 00B8	¹ 00B9	º 00BA	» 00BB	¼ 00BC	½ 00BD	¾ 00BE	¿ 00BF
C0	À 00C0	Á 00C1	Â 00C2	Ã 00C3	Ä 00C4	Å 00C5	Æ 00C6	Ç 00C7	È 00C8	É 00C9	Ê 00CA	Ë 00CB	Ì 00CC	Í 00CD	Î 00CE	Ï 00CF
D0	Ð 00D0	Ñ 00D1	Ò 00D2	Ó 00D3	Ô 00D4	Õ 00D5	Ö 00D6	× 00D7	Ø 00D8	Ù 00D9	Ú 00DA	Û 00DB	Ü 00DC	Ý 00DD	Þ 00DE	ß 00DF
E0	à 00E0	á 00E1	â 00E2	ã 00E3	ä 00E4	å 00E5	æ 00E6	ç 00E7	è 00E8	é 00E9	ê 00EA	ë 00EB	ì 00EC	í 00ED	î 00EE	ï 00EF
F0	ð 00F0	ñ 00F1	ò 00F2	ó 00F3	ô 00F4	õ 00F5	ö 00F6	÷ 00F7	ø 00F8	ù 00F9	ú 00FA	û 00FB	ü 00FC	ý 00FD	þ 00FE	ÿ 00FF

ISO Character Set 8859-1 (Latin 1)

Appendix C Installation

Installation Notes

> *Note:* *Evaluation copies of RFTerm allow the user to Connect eleven (11) times. The LXE Software License Agreement dialog appears after each connect request and before the connection is activated. Select either Accept or Reject to continue. When the counter reaches 0, you cannot use the RFTerm Connect function. Contact your LXE representative for assistance when using RFTerm evaluation copies or when the evaluation counter expires.*

Windows 2000 / XP

If your device is using Windows 2000 or Windows XP, the following steps must be taken to ensure RFTerm can read and write to the registry before initial installation or upgrading to a newer revision:

- Log on as the Administrator.
- Open the registry editor.
- Delete the entire registry key HKEY_LOCAL_MACHINE\SOFTWARE\LXE\RFTerm.
- Do *not* delete the LXE key or any other items under it except RFTerm.
- Browse to the location of the `RFTerm.inf` file, click on the file and select **File** | **Install**. Follow the on screen prompts.
- Launch RFTerm for the initial time as the administrator.
- This will create the RFTerm registry with guest privileges.

Windows CE

When you upgrade RFTerm to a new release version, you must begin with a clean registry. If your device is using Windows 2000 or Windows XP, use the procedures listed above. On all other devices, use the following steps:

- Exit RFTerm if it is currently running. (On Pocket PC devices, go to Start | Settings | System | Memory | Running Programs. If RFTerm is on the list, highlight it and press "Stop".)
- Open the registry editor.
- Delete the entire registry key HKEY_LOCAL_MACHINE\SOFTWARE\LXE\RFTerm.
- Do *not* delete the LXE key or any other items under it except RFTerm.
- Launch the RFTerm cab file.
- Launch RFTerm to complete the installation.

Revision History

Revision A, July 2002, Initial Release

Revision B, October 2002

Section	Explanation
Entire Manual	Revise contents to reflect full version software enhancements.

Revision C, April 2003

Section	Explanation
Entire Manual	Revised contents to reflect functional changes re Microsoft® Pocket PC® device users. Removed sections titled "Printers", "Command Line Options" and "Troubleshooting."
Appendix B – Key Maps	Added RFTerm, ANSI Plus, IBM 3270, IBM 5250 key maps for the LXE MX3-CE and MX5 devices.

Revision D, June 2003

Section	Explanation
Entire Manual	Revised contents to reflect functional changes for Microsoft® Pocket PC® device users. Removed sections titled "Printers", "Command Line Options" and "Troubleshooting."
Appendix B – Key Maps	Added RFTerm, ANSI Plus, IBM 3270, IBM 5250 key maps for the LXE MX3-CE and MX5 devices.
Entire Manual	Added ANSI capabilities. Added Windows OS specific notes.

Revision E, January 2004

Section	Explanation
Chapter 1 – Introduction	Deleted "Macro Record and Playback" and "Integrated Demo Modes".
Chapter 2 – Quick Start	Deleted "Automatic Navigation" and "Auto-start Sessions".
Chapter 3 – RFTerm Settings	Changed title from "RFTerm Configuration". Moved "Connection", "Colors" and "Barcode" sections to Chapter 4. Updated images: Configure 3270 Host Connection, Configure VT220 Host Connection, Configure 5250 Host Connection, Configuration – Display Tab, Configuration – Colors Tab, Configuration – Cursor Tab, Configuration – Barcode Tab, Barcode Character Mapping, Configuration – Options Tab, Added "Retry on Signal Loss", "Send Key", "Serial Data – XP/2000 and CE. NET Only", "Auto Login", "Function Key Programming". Deleted sections "Advanced", "Auto Connect", "Auto Reconnect" and "Macro on Connect", "COM Port". Corrected menu path statements. Moved "Font" to "Color" section.

Chapter 4 – Session Configuration	Changed title from "Session Interaction". Deleted "Auto-Start a Host Session", "Macros", "Macro Status", "Delay", "Recording", "Playback", "Automatic Login", "Automatic Host Login". Added "Out of Range". Updated "Options" section with new parameters. Updated "Character Mapping" and "Autologin" sections. Changed "Serial Data …" section to include references to Windows CE. NET devices.
Chapter 5 – ANSI	Renumbered as Chapter 7.
Chapter 5 – Session Interaction	Created new chapter containing "Chapter 4 – Session Interaction" contents from revision D.
Chapter 6 – 5250	New chapter describing and explaining the LXE 5250 Terminal Setup Print Command.
Chapter 7 – ANSI	Renumbered Chapter 5 from revision D. Added "DECRM", "DECSM", "PL" and "PS" sections.
Appendix A – Key Maps	Updated "Icon/On-line Key Identification". Changed miscellaneous path and keypress indicators. Added RFTerm, ANSI Plus, IBM 3270, IBM 5250 key maps for the LXE MX6 device.
Appendix B – ANSI Quick Reference Guide	Added "Printing" section.

Revision F, February 2004

Section	Explanation
Chapter 1 – Introduction	Edited "Smart Pads" section.
Chapter 2 – Quick Start	Updated Figures in section "RFTerm Screens on Different Sized Platforms". Edited section titled "Predefined Hotkeys".
Chapter 3 – RFTerm Configuration	Added hotkey sequence for "Hide O1A" section. Corrected "Data Bits" and "Stop Bits" sections.
Chapter 5 – Session Interaction	Updated "Text Input Tool Display" figure. Updated "Access Control/Device Lockdown" section. Fine tuned "Hotkey Shortcuts" default hotkey list. Removed "Online Help" section.
Chapter 6 – ANSI	Removed the word "Plus" from "ANSI Plus" throughout the reference guide.
Appendix A – Key Maps	Updated "RFTerm Functions", "MX3-CE Key Maps", "MX5 Key Maps", "MX6 Key Maps", "VX4 Key Maps (was XP/2000)" sections. Added section "VX4 CE .NET".

Revision G, March 2004

Section	Explanation
Chapter 2 – Quick Start	Edited section titled "Predefined Hotkeys".
Chapter 5 – Session Interaction	Fine tuned "Hotkey Shortcuts" default hotkey list.
Chapter 6 – ANSI	Removed BEL control code note.
Appendix A – Key Maps	Added sections "VX4 CE .NET" and "VX5 Key Maps".

Revision H, June 2004

Section	Explanation
Chapter 1 – Introduction	Added description for Help key sequence feature.
Chapter 4 – Session Configuration	Added Auto-Transmit feature in "Connection". Updated screen shots to show the Auto-Transmit check box. Updated Auto Login section to reflect 6 pairs of ANSI prompts/replies.
Chapter 5 – Session Interaction	Added the Help hotkey sequence to the Hotkey Shortcuts table.
Appendix A – Key Maps	Added section for the MX3X keypad.

Revision J, August 2004

Section	Explanation
Chapter 1 – Introduction	*Added*: Automatic Session Persistence, Double Byte Support, Installation Notes, *Updated*: Access Control, Programmable Function Keys, Printing, Help. *Removed*: Host Session Password
Chapter 2 – Quick Start	*New graphics*: Figure 2-1 to 2-4 *Updated*: Quick Start, Application ToolBar.
Chapter 3 – RFTerm Settings	*New graphics*: Figure 3-1, 3-2, 3-5, 3-6, 3-7, 3-8. *Added*: Hide MenuBar, Shortcuts, Options, Serial Data. *Updated*: Hide All Bars, Hide ToolBar, Config Password, Enable HotSpots, Config Keys, KeyBar, Custom Bar. *Removed*: Auto Enter.
Chapter 4 – Session Configuration	*New graphics*: Figure 4-1 to 4-3, 4-14 to 4-16. *Added*: "3270, 5250, VT Config", Auto Reconnect, VT Configuration - Keys, VT Configuration – Screen, VT Configuration – Options, VT Configuration – Answerback, 5250 Configuration, 3270 Configuration, Font, Colors, Use Field Exit. *Updated*: Character Mapping, Function Key Combo Box. *Removed*: Auto Transmit, Modes.
Chapter 5 – Session Interaction	*New graphics*: Figure 5-1 to 5-11. *Added*: Escape, Roll Up, Roll Down, Select, Print, Break, Cancel, Answerback, Escape, Backspace, Line feed, COM Port Error. *Updated*: Password Protection, 3270 Host Key Descriptions, 5250 Host Key Descriptions, VT Host Key Descriptions, Function Keys, Print Setup, Text InputTool, HotSpots, HotKey Shortcuts, Automatic Session Startup. *Removed*: Loading a Session Configuration, Savings a Session Configuration, "PA1, PA2, PA3".
Appendix A – Key Maps	*Updated*: Icon / Button Identification, RFTerm Functions, RFTerm Functions and ANSI Functions for the : MX3-CE, MX3X, MX5, MX6, VX4 (XP/2000), VX4 (CE.NET), VX5.
Appendix C – Installation	Added new appendix.

Chapter 3 – RFTerm Settings	Added sections: "Beep on Error", "Toolbars, Keybars, Shortcut Keys and the MX6" and "MX6 35-key Buffering".
	Modified sections: "Introduction", "Shortcuts", "Ctrl Arrow Scrolling keys", "Sounds", "COM Port" and "Stored Forms".
	Deleted sections: "Cursor" (moved to Chapter 4),
	Renamed "Config Keys" to "KeyBars".
Chapter 4 – Session Configuration	Added a: "Compatibility" section to "VT Configuration", "5250 Configuration" and "3270 Configuration".
	Added sections for new tabs: "Cursor", "Print Data Map" and "Keybd".
	Added sections for new parameters: "KeepAlive", "Use LXE Format", "Pointsize", "Specify Rows and Columns", "Lock Position", "Disable Blank Keys", "DEC Special Graphics" and "Multi beep on Good Scan".
	Modified sections: "Introduction", "Telnet Port", "Local Echo", "Auto Transmit", "TN3270E", "Character Stripping" and "Auto Login Reply".
Chapter 5 – Session Interaction	Added section: "Device Name in Use".
	Added sections for new parameters: "Installed Printer", "Direct Serial", "Bluetooth", "Scanner Applet" and "Pass thru CR".
	Modified sections: "Print Setup", "Out of Range", "Hotkey Shortcuts", Automatic Session Startup", Direct Serial" and "Hotspots".
Chapter 6 – ANSI	Added sections: "DECDHL", "DECDWL", "G0/G1/G2/G3 – Designate G0/G1/G2/G3", "SI – Shift In (Lock Shift G0)" and "SO – Shift Out (Lock Shift G1)".
Chapter 7 – 5250	Added Sections: Input Device ID, Custom Character Mapping, Custom Keyboard Mapping.
Appendix A – Key Maps	Removed VX4 and MX5 Pocket PC Key Maps. The VX4 family of devices is obsolete. The MX5 Pocket PC 2000/2003 is obsolete. Added VX3X to MX3X segment.
Appendix C – Installation	Added note directed to user of an RFTerm evaluation copy.
	Added Revision History section.
	Modified "Installation Notes" section.

Revision N, May 2007

Section	Explanation
Chapter 2 – Quick Start	Added HX2 splash screen.
Chapter 4 – Session Configuration	Added HX2 to "DoubleByte Fonts and Mobile Clients".
Appendix A – Key Maps	Added HX2 Key Maps.

Revision P, November 2007

Section	Explanation
Chapter 1 – Introduction	Added new section: "About RFTerm"
Chapter 2 – Quick Start	Added MX8 to "RFTerm Screens on Different Sized Platforms".
Chapter 4 – Session Configuration	Added new section: "HX2 Function Key Enable".
Appendix A – Key Maps	Added the following key maps: 　　Virtual key maps 　　HX2 key maps for new keypad styles 　　MX8 keymaps. Revised MX7 keymaps for Scroll Left One Row and Scroll Right One Row.

Index

D

E

N

O

M

P

T

U

V

W

Z