A N Z I O

Personal Computer Communications

User Manual

Version 17.0b – December 2013
Copyright © 1987–2013 by Robert Rasmussen
All Rights Reserved

Rasmussen Software, Inc.

10240 SW Nimbus Ave.   Suite L9   Portland OR 97223   USA
(503) 624-0360   www.anzio.com   RSI@anzio.com

Acknowledgments

Windows is a registered trademark of Microsoft Corporation

Other product names are trademarks of their respective owners.

Printing History

Edition 8.6r   March 1988
Edition 9.0   November 1988 (Second printing December 1989)
Edition 9.5   February 1990
Edition 9.7   March 1993
Edition 9.8   June 1993
Edition 10.9   July 1996
Edition 11.0   November 1996
Edition 11.4   March 1998
Edition 12.0   November 1999
Edition 12.5f   October 2001
Edition 15.0q   April 2004
Edition 15.0u   December 2004
Edition 15.1   March 2005
Edition 15.2   August 2006
Edition 16.0c   January 2008
Edition 16.0h   March 2008
Edition 17.0b   December 2013

Table Of Contents

Part IV   Technical Reference Guide 198

Introduction

Anzio is a family of personal computer (PC) communications programs. Anzio provides *terminal emulation* (also known as *telnet*) for a large set of terminal types and operating systems. That is, with your PC connected to a host system, Anzio manages the communication to appear exactly like a standard "dumb" terminal.

Anzio also provides "smart" terminal functions including *passthrough print*, macros, and several types of file transfer.

- To install and run Anzio, see *Installing and Using Anzio*, page 10.

- To stop and exit Anzio, click on the File menu, and then the Quit item.

> ➢ If you are migrating from an earlier version of Anzio, check Appendix C, *Migration From Earlier Releases*, page 233.

While Anzio is running, the PC behaves very much like the terminal it is emulating. The host system thinks it has a terminal attached, and displays information accordingly.

Pressing a keyboard key sends one or more characters to the host. Function keys (F6) and special keys (PG DN) are usually configured to send certain control-character sequences to the host system.

- Some key combinations are predefined by Anzio to control Anzio's operation. Anzio allows you to change the definition of any key. Keys can also be defined to perform a series of Anzio commands and operations – see page 37.

Anzio sometimes uses the line at the bottom of the terminal emulation window to accept commands and display messages and status information. In general, anything you type on the bottom line goes to Anzio, and is not sent to the host.

The Anzio family currently consists of two members, AnzioWin and Anzio Lite. Both are 32-bit applications, that run on Windows 98SE through Windows Vista. Older members of the family included versions of AnzioWin and Anzio Lite for 16-bit platforms (Windows 3.x), as well as versions for DOS.

Key Features

Anzio's primary feature is reliable, fast terminal emulation. Other key features are listed here. Note that some features are in AnzioWin only, not in Anzio Lite.

Multiple emulations	Anzio emulates many popular terminal types: VT100/220/320/420, SCO ANSI, AT386, Wyse 50, Wyse 60, Versyss C332, and ADDS Viewpoint, among others.
Terminal features	Graphics character sets, answerback, downloadable keys, downloadable characters, status lines, video attributes, color, passthrough print.
Connection methods	Anzio can connect to the host via SSH, telnet, rlogin, modem, direct serial, or PicLan.
80- and 132-column screens	Anzio provides 80- and 132-column screens, and custom screen sizes.
Screen control	Run in full-screen mode, with no extraneous screen clutter, or in a smaller window.
Toolbar	A toolbar can be configured and displayed, with each toolbutton executing a command or a macro.
Security	Anzio supports SSH1, SSH2, and SRP, for secure encrypted communication. Passthrough print data will also be encrypted if these protocols are used.
Login Wizard	If Anzio knows your username and password, it can handle the login for you.
Language support	Anzio can display and send characters for many languages. Anzio uses 16-bit Unicode characters internally. Unicode defines over 39,000 characters used in European, Hebrew, Arabic, Chinese, Japanese, Korean, and other languages. If your

host system provides language support, Anzio can display and print your language.

International keyboard support

Anzio properly supports character entry from any Windows keyboard. In addition, international characters can be entered by using deadkeys, alt-numeric entry, Unicode values, etc.

Diacritic support   Anzio has special treatment for combining diacritics, especially those used in the MARC standards for libraries.

Font coverage   Anzio automatically finds installed fonts to display special and international characters.

Linedraw characters The characters used to draw boxes and other lines on terminals are always displayed properly, connecting to adjoining lines.

Browser integration Anzio can register itself as the standard telnet client in Windows, so that clicking on a telnet cross-reference in a web page will invoke Anzio and create a telnet session to the indicated host. The same is true for SSH.

Mouse support   Various mouse button actions can be configured to do different things, including sending screen data to the host, launching a URL, copy and paste, and more. The mouse wheel can control screen scrollback (review) or send arrow-key sequences to the host. Left-tilt and right-tilt on the wheel can have macros assigned under Vista.

Mouse-to-host   Mouse-to-host protocols as in xterm and VT420 are supported, so host-end programs can work with Anzio to receive mouse events.

Sound   Anzio can issue a beep, play a standard Windows sound, or play WAV and other media files from the local PC or the Internet. Beep signals from the host can generate sounds from the motherboard speaker or the sound card.

Pictures   Anzio can display a bitmap image, for screen watermarks, text-over-forms, pictures of products or employees, etc.

Image box	Anzio can show a second, floating window for an image, such as an image of a document. Several file formats are supported, including image-PDF and PCL.
AUX serial devices	Anzio can accept input from auxiliary serial devices, such as barcode readers, postage scales, etc. It can also do bidirectional communication with AUX devices, such as for point-of-sale support.
Backchannel serial	Under an SSH connection, Anzio can make PC serial ports appear as virtualized ports on the server.
Kiosk mode	Anzio can be run with a limited menu system, disabling unneeded user commands, for situations such as public-access PCs in libraries.
Lockdown	Anzio can be locked down in other ways, so users can't save any changes, can't exit improperly, etc.
Screen review	Anzio buffers data that scrolls off the top of the screen, including cleared screens. You can later scroll this data back into view. Buffer size is configurable.
Key mapping	You can control what code sequence is sent out by various special keys, including 48 function key combinations, control-keys, alt-keys, and special keys. Numeric pad keys can be mapped separately in NumLock on and NumLock off states.
Application key support	
	Anzio can capture and react to special application keys on the keyboard, such as "Print" and "Paste".
Macro language	Assign user macros to unused keys. Macros can send common keystroke sequences to the host, or use Anzio's more than 300 commands. A macro can be run on program startup. Macros can be associated with mouse events and other events.
Control from host	Programs running on the host system can control and configure Anzio and its PC, including PC file operations, DOS commands, Windows operations, and local printer control. This capability is subject to security settings in Anzio.

Local operations	Anzio provides many local PC file operations, such as DIR, RENAME, TYPE, LOG, etc. Anzio can initiate any other Windows or DOS program, optionally waiting for it to finish.
Screen capture	Capture incoming data to disk file, with various options for handling control characters. International characters can be written in any of three character sets, including Unicode UTF-8.
File transfer	Anzio provides a variety of file transfer protocols for different host systems and file types, including Kermit, Xmodem, Ymodem, and Zmodem
Interactive FTP	Anzio provides a point-and-click FTP client for transferring files to and from a host system.
Programmed FTP	Anzio provides a command-driven FTP client, allowing fully automated file transfers.
SFTP	Anzio provides both point-and-shoot and command-driven SFTP (Secure File Transfer Protocol) operations when connected via SSH.
Copy-and-paste	Anzio can copy part or all of the screen onto the Windows clipboard (text, rich text, and bitmap formats), and can paste clipboard text to the host program, with full international language support.
Ink markup	On TabletPCs and most Windows XP systems, you can add "ink" markup to the main window, and then copy-and-paste that into word processing software for user documentation, for instance. Optional shape recognition converts hand-drawn rectangles (for instance) to perfectly straight rectangles. The ink is also included in screen prints.
Gesture recognition	In TabletPCs, the user can use "gestures" with the stylus to input common special keys, such as <enter>, <backspace>, <Esc>, etc.
Configuration	You can configure many "comfort" features, such as colors, mouse click actions, non-blinking cursor, status/gauge line, beep pitch, etc.
Scripting	Anzio can invoke scripts in VBScript or JavaScript, with data exchange.

DDE client	Anzio can act as a DDE client, and send commands to other programs.
DDE server	Anzio can act as a DDE server, so that an external program can control its operation.
OLE client	Anzio can act as an OLE client, and send commands to other programs.
Languages for menus	Choose to display Anzio's menus, prompts, etc. in Spanish, French, etc.
Scanner support	A scanner or other device with a TWAIN driver can be used as an input file for printing, file transfer, etc.
Speech output	Part or all of the screen can be read out loud via Windows' text-to-speech capability.
Drag-and-drop	Anzio can be configured to do a specified operation when something is "dropped" on the main window.
Emailing	Anzio has an interface to four different email protocols, allowing interactive or programmed emailing.
Local port forwarding	Under SSH connections, local PC ports can be forwarded to the server.

And then there's printing. Anzio has so many features for printing they require a separate list:

Print screen	Anzio can print the current screen (even with the PRTSC key). It can also print just a portion of the screen.
Print a file	Anzio can print a file that exists on the PC, on the network, or on an FTP or HTTP server.
Passthrough print	Anzio can print print jobs sent from the host in the "transparent print" or "local print" protocol.
Capture to printer	The operator can cause data coming to the screen to be sent to the printer.
Print levels	Anzio can send data to the printer at 4 different levels, for different print requirements. Printing can use the Windows printer driver, or can bypass it.

Auto-fit	The Print Wizard feature will analyze each print job and automatically fit it to the page. Wide format jobs can be printed in landscape mode automatically.
Mini-page	For one-at-a-time address labels, receipts, and similar jobs on continuous feed paper, Print Wizard can create a custom page length, just as long as is needed.
Multi-column	Plain text input can be converted to multi-column mode, for labels.
Multilingual	The full range of Unicode text can be printed, with special handling of linedraw characters and diacritics, and automatic font selection.
Forms overlays	Bitmap or PCL forms can be added to print jobs.
Scanner support	Scan a file as an overlay, or include it in a print job.
Barcodes	Fifteen kinds of barcodes can be printed on almost any printer.
Markup language	The Print Wizard Markup Language gives you the means to direct all kinds of fancy printing, including font changes, rectangles, pictures, paper size, and more.
Print to PDF	The Print Wizard feature has a built-in PDF generator, so any print job can create a PDF file.
Print to TIF	Print jobs can be rendered as TIF files, in configurable format.
Print-and-email	Print jobs that go to PDF or TIF can be emailed, either automatically or with user assistance.
Print to fax	Print Wizard can output to WinFax Pro or the fax engine included in later versions of Windows.
PCL translation	Print jobs containing PCL codes can be printed on any printer, or sent to fax, or translated to PDF files.
Backchannel print	Under an SSH connection, extra channels can be opened for printing from the host.

Organization of this Manual

This manual is organized in five parts:

- Part I, Narrative, describes Anzio concepts, operations, and usage. Please humor us and read Part I before you call with questions.

- Part II, Data Capture and File Transfer, describes how to use Anzio to capture host data off the screen, and how to transfer files between your PC and different host operating systems.

- Part III, Command and Control, describes all Anzio commands in alphabetical order, followed by the startup options for the Anzio program itself.

- Part IV, Technical Reference, provides some byte-level specifics on communication protocols, sending Anzio commands from the host, terminal emulation, and data capture.

- Part V, Appendices, provides additional information on Anzio and various hosts:

    - Appendix A, *Installing and Using Anzio*
    - Appendix B, *Notes On Particular Host Systems*
    - Appendix C, *Migration From Earlier Releases*
    - Appendix D, *Serial Communication Problems*
    - Appendix E, *Distribution Information*

Conventions

- "Anzio" refers to any version of Anzio; that is, Anzio Lite or AnzioWin.

- "PC" is the personal computer running Anzio.

- "Host" is the computer (server) to which Anzio is connected.

- ⏎ represents the "Carriage Return" character, generated by the "Enter" key on the PC keyboard.

- F6 represents function key F6 on the PC keyboard.

- "Communicate/Hangup" indicates the Communicate menu, and the Hangup menu item.

Part I   Narrative

1   Installing and Using Anzio

This section describes how to get the Anzio program up and running:

- *Installing Anzio*
- *Making the Connection*
- *Starting Anzio*
- *Setting Default Anzio Communication Parameters*
- *Initial Anzio Terminal Screen*
- *Logging On to the Host System*
- *The Anzio Window*
- *Exiting from Anzio*
- *Typical Terminal Types*
- *Using Function Keys and Special Keys in Terminal Emulation*
- *Anzio's Mouse Support*
- *Selecting Text Using Keystrokes*

➢ The README.TXT and HOWTO.TXT files on the distribution disk provide additional information about Anzio installation and usage.

1.1   Installing Anzio

Anzio can be installed from the distribution CD or from the web site. Anzio cannot be run directly from the distribution CD, because those program files are compressed.

If you copy the installation file from the CD or web site to another location, you can also run it from there. This location can be any accessible disk, including network drives and removable USB drives ("thumb drives").

Installing Anzio consists of:

- Transferring the program files to your PC.
- Building Start menu items to initiate Anzio.
- Building one or more desktop icons to initiate Anzio.
- Placing certain information in the Windows registry

For more information on the contents of the distribution CD, and the system requirements for Anzio, see Appendix E, *Distribution Information*, page 235.

> ➢ After installing the Anzio executable, you may customize it, by using a Windows resource editor to add or remove menu items and accelerator keys – see Editing the Resources, page 220.

1.1.1   Installing Anzio on the PC

Anzio is available on CD-ROM, and also on the Internet at http://www.anzio.com. Both methods use standard Windows installation techniques. In both cases, the installed Anzio Lite or AnzioWin will run as a demo, for up to 60 days, until and unless you enter your serial number and license code as delivered by Rasmussen Software or your software supplier.

To install from the web site:

1. Point your web browser to the appropriate page at the web site shown above.
2. Click the "download" item.
3. Depending on your browser's capabilities, you may be able to run the installer directly from the web site. Otherwise, save the file to disk, then run it.

To install from CD:

1. Insert the distribution CD into any available CD or DVD drive.
2. The Setup program may start automatically; if not, continue…
3. Start an Explorer window and navigate to the disk drive.
4. Double-click on the SETUP.EXE program.

1.1.2   Installing Files on the Host

For normal connection and terminal emulation, you will not need to do anything to your host system. However, the Anzio distribution disk in-cludes some files for use on the host computer, to a) better support the 'Anzio' terminal type, b) provide some kinds of file transfer, and c) support certain application programs that run on the host. Because these files are generally **not** needed, we'll address them later.

1.1.3   Uninstalling Anzio

To uninstall Anzio, just go to the "Add/Remove Programs" item in the Windows Control Panel, and select Anzio. The uninstall process will leave behind certain files that you have created, such as settings files and keys files.

1.2   Making the Connection

Anzio can communicate with the host system in several ways, that reduce down to a) network, b) modem, and c) direct serial. Note that if you are connecting your PC to the Internet through a modem connection (analog, cable, or DSL) to an Internet Service Provider (ISP), or to another network through a modem ("PPP dialup"), Anzio considers this a *network* connection. This is an important distinction; if you are not sure what kind of modem connection you ought to be using, check with your system administrator.

The following sections will help you get your connection in place.

1.2.1   Local Area Network (LAN)

If your PC is going to communicate over a local area network, simply plug the network cable into your network interface card. You will also need to configure TCP/IP at the Windows level. That is beyond the scope of this manual.

1.2.2   Permanent Internet Connections

If your LAN is connected to the Internet through a router, a cable modem, a DSL modem, etc., you'll be able to reach any server on the Internet, as well as local servers.

1.2.3   Wireless Internet

If your wireless Internet adapter has connected through an access point or router to either a local area network or the Internet, Anzio can connect over that network.

1.2.4   Internet Dial-up

You may need to connect to the Internet through an ISP, by telling Windows to dial your modem. Again, you'll need to have that configured already. And you'll need to have that connection established before you run Anzio.

1.2.5   Analog Modem

To connect your PC as a terminal, through your PC's modem to a modem on a host system, you'll obviously need a modem on your PC, with its Windows driver installed, and its cord plugged into a live telephone jack. In this situation, Anzio will initiate the dialing operation.

> ➢ A DSL modem or a cable modem is different. See above.

1.2.6   Direct Serial Connection

This section applies only to a direct serial connection from your PC to the host system's serial port. This is often done where a dumb terminal has been replaced by a PC running Anzio.

The physical hookup between your PC and your host machine is usually a standard terminal cable. The host machine sees your PC as a plain terminal (CRT). The cable coming from the host is the same for a CRT as for Anzio.

For the initial installation test, we recommend that you unplug a line from a working CRT and connect that line to the Anzio PC. This way you'll know that the line works and is properly configured, eliminating one potential source of problems.

1.2.6.1   Configuring the Serial Port

Your PC must have a *serial port* (*comm port*). This may be on the mother board, on an add-in board, or on a USB-to-serial adapter. The port must be installed to Windows, with no hardware conflicts, and operational.

The configuration of the serial port (baud rate, etc.) is all handled by Anzio. Settings in the control panel are ignored.

1.2.6.2   Serial Cable Wiring

The PC serial port always has a <u>male</u> plug, usually 9-pin, sometimes 25-pin. A 25-pin <u>female</u> plug in the back of a PC is probably a parallel printer port - don't use it for a serial connection.

If the cable from your host system ends in a male plug, you will need a 25-pin *female-to-female gender converter*, available from any PC sup-plier.   In addition, if your cable terminates in a 25-pin plug, and your PC has a 9-pin plug, you will need a standard 25- to 9-pin adapter.

> ➢ Do not use a "null modem" adapter.

The standard 9-pin wiring connection is:

1	CD (carrier detect)
2	SD (send data)
3	RD (received data)
4	DTR (data terminal ready)

5	GND (ground)
6	DSR (data set ready)
7	RTS (request to send)
8	CTS (clear to send)
9	RI (ring indicator)

The standard 25-pin wiring connection is:

2	SD (send data)
3	RD (received data)
4	RTS (request to send)
5	CTS (clear to send)
6	DSR (data set ready)
7	GND (ground)
8	CD (carrier detect)
20	DTR (data terminal ready)
22	RI (ring indicator)

1.3   Starting Anzio

Start Anzio from its icon on your desktop, or use the Windows Start menu – Start/Programs/Anzio xxx/AnzioWin (default connection) (or Anzio Lite).

Anzio will prompt for communication parameters, described in the next section.

1.4   Setting Default Anzio Communication Parameters

The first time you run Anzio, you must set several communication parameters:

- *Terminal Type*
- *Communication Method*
  - *Network Parameters* – Type, Host name or address
  - *Serial Parameters* – Port, Baud rate, and data bits and parity.

Anzio shows three dialogs, the first to select terminal type, the second showing the Anzio version and any files used, and the third for selecting communication parameters:
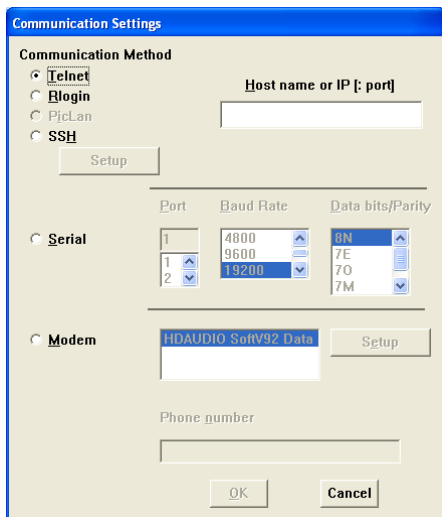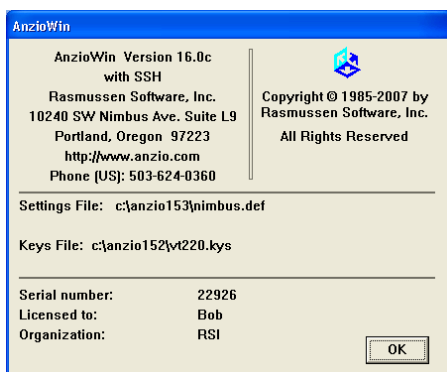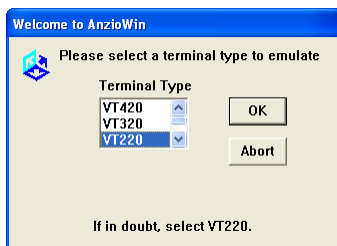
The parameters you enter here will be saved in a *defaults file* (also known as a *settings file*), normally named "AnzioWin.def". The settings file also contains all the other configuration settings for Anzio, except for key mappings.

When you exit Anzio, it will ask if you want to save changes to the settings file. The next time you run Anzio, it reads the settings file to set the default system settings. You may change these settings at any time with Anzio commands or menu items.

> ➢ The settings file is in text format, and can be edited with Notepad, for instance.

1.4.1   Terminal Type

Anzio can emulate many different terminal types. The host is probably expecting a particular terminal type. Select that terminal type to start. If in doubt, select "VT220". Later, you can change the current terminal type with Communicate/Terminal Type or the TERM command.

See *Typical Terminal Types*, page 24, for more information on choosing a terminal type.

1.4.2   Communication Method

Anzio needs to know how your PC will communicate with the host.

- ▪ If you do not know some of these parameters, ask a local expert for help.

1.4.2.1   Network Parameters

A network connection requires a previously installed Windows interface. Anzio checks to see which of the interface types are installed, if any, and enables the corresponding radio buttons.

- ▪ Select the network interface type (Telnet, Rlogin, SSH, etc.).
- ▪ Enter either the host name, or an IP (*Internet Protocol*) address, for the host (server) you're connecting *to*, **not** for this PC.
- ▪ If you need to connect to a non-standard port, follow the host name or IP with a colon (":") and a host port number (the standard port for telnet is 23; for SSH, 22).
- ▪ If you choose SSH, you can configure the SSH parameters, by clicking the Setup button next to "SSH".

1.4.2.2   Serial Parameters

If you choose a serial (comm) port, you will need to configure the serial communications. Note that this configuration overrides anything set in the Windows Control Panel.

The following parameters apply to direct serial connections:

Port   Anzio needs to know which serial port is connected to your host computer – see *Configuring the Serial Port*, page 13.

Select (or enter) the PC's serial port number. If no serial ports are shown, then Windows is not finding any serial ports installed.

Baud Rate   Your PC and your host machine must communicate at the same speed, known as the *baud rate* (data bits per second). The host's baud rate is generally preset, and Anzio must match it.

Select the baud rate at the prompt.

Data Bits/Parity   Each "byte" sent or received has some *data bits*, an optional *parity bit*, and one or two *stop bits*. The parity can be odd, even, mark, space, or off. Typical configurations have one stop bit and: 8 data bits/no parity (8N), 7 data bits/parity bit even (7E), or 7 data bits/parity odd (7O). Select these parameters to match the host.

Use the DATA BITS, PARITY, and STOP BITS commands to set nonstandard configurations.

1.4.2.3 Modem Parameters

Select the modem (if your PC has more than one). You may also want to configure the modem, by clicking the nearby Setup button.

> ➢ This area applies only to modem-to-modem connections, not modem-to-network connections.

Then enter the phone number to be dialed. As soon as you click "OK", Anzio will tell the modem to dial.

1.5   Initial Anzio Terminal Screen

By now you should be looking at a blank screen with a *status line* at the bottom. Anzio also has a typical menu bar at the top (File, Edit, View, etc.), and it may have a *toobar* at the top.

> ➤ On an SSH connection, you will need to go through the login process *before* you get to the main screen.

The status line shows status indicators (of course) and some error messages. You can turn the status line on and off with View/Show Status Line or the STATUS LINE command. The status indicator words and their meanings are:

CAPS	The ⌨CAPS LOCK key is on – all alphabetical characters will be in UPPER CASE.
NUM	The ⌨NUM LOCK key is on – the numeric keypad sends digits, rather than the other codes on those keycaps.
LOCK	The keyboard is locked – nothing you type will go through to the host. To send keyboard keys, either the host must unlock the keyboard, or you must use Communicate/Unlock, ALT U , or the LOCK OFF command.
HOLD	The output from the host to Anzio has been suspended, with either the HOLD command or the SCROLL LOCK key.
AUX	Indicates the aux port is enabled for connection to a serial device, such as a barcode reader.
PAGE *n*	You are viewing page or screen *n* in a multi-screen environment.

From the live terminal screen, any "normal" key you press will go out through the communication line to the host computer. Function keys and special keys usually send their own terminal-specific codes. However, you can define almost any keyboard key to send special character sequences or to call Anzio functions.

The status line can also be used to enter Anzio commands, called with CTRL SHIFT F . We'll get into that later. For now, let's see if we can communicate.

1.6   Logging On to the Host System

If you are connected to a UNIX (or similar) system, but haven't seen a login prompt, press ⏎. You should see the "login:" prompt (or "user" or "username" or …). Enter your login name and password just as you would with a terminal. Watch out for uppercase vs. lowercase.

After you log on, you will see the operating system's "banner" at the top of the screen. The host system may take you directly in to an application, or ask you some questions, or drop you at a "shell" level. Try typing the appropriate keys, to check the connection.

If something goes wrong at this point, see the "Troubleshooting" section below.

To use Anzio menus and enter commands, see *Talking with Anzio*, page 32. The *Command Reference* section, starting on page 94, describes all commands.

1.6.1   Anzio's Login Wizard

For future connections, Anzio can login for you automatically (for most kinds of connections). To do so, it must know your login and password. If you are interested in using this feature, proceed as follows.

Go to Communicate/Login/Username, and enter your username. Go to Communicate/Login/Password, and enter your password. Finally, in Communicate/Login, make sure Login Wizard is checkmarked.

Quit from Anzio. When it asks if you want to save your settings, answer "Save". The next time you run Anzio, it will attempt to login for you.

> ➢ Login Wizard will not work on direct serial connections, for technical
>   reasons. It is disabled on SRP connections, for security reasons. In
>   SSH connections, its use is governed by settings in Edit/Advanced
>   Options/Security.

> ➢ The username is stored in the settings file in plaintext. The password
>   is stored in a rather simple encoded format. This may constitute a
>   security risk.

1.6.2   Troubleshooting

This section describes some common problems and typical solutions. Note that you can obtain more connection diagnostics by doing Diagnose/Debug communications.

1.6.2.1   Trouble on Network Connections

If TCP/IP networking is properly configured, and the host system is accepting connections of the type you're trying, Anzio should connect.

If you get an error message "WINSOCK ERROR 10065", it means your TCP/IP networking is not set up on this PC.

If you get a "Connection refused" or a "Connection timed out" message, your host system is not allowing you in. Check your host name or IP address. Make sure the host allows this kind of connection. Make sure any user limits have not been reached.

It is also possible that your connection is hitting an external firewall or router, or an internal (software) firewall, that does not let this type of connection occur. If this is true, you will need to configure that software to either a) allow Anzio as a trusted application, or b) allow an outbound connection on the port in question (typically 22 for SSH, 23 for telnet).

For other network problems, including other Winsock errors, test your TCP/IP network with the "ping" or "tracert" utility, or try running telnet.exe that came with Windows, and connecting to the same host.

1.6.2.2   Trouble on Direct Serial Connections

If you have gotten no response at all, you have a problem with:

- your PORT or BAUD setting (change with Communicate/Setup or the corresponding command),
- your serial port hardware, or
- your cabling.

If you get "garbage" on the screen, chances are that your BAUD rate is incorrect. Try another baud rate, using Communicate/Setup.

If you see "foreign" characters, you may need to change parity and data bits with Communicate/Setup or the corresponding command.

1.6.2.3   Trouble on Modem Connections

If the modem fails to connect, or if you get a "TAPI call state error", it can be hard to diagnose. The answering modem may have been busy, in which case you just need to try again. Try dialing the same phone number with a regular telephone – you should hear the modem answer and squeal. If you hear a recording, or a person answers, that will tell you the problem. Or, plug an analog phone into the same line as the modem is on (with a 2-phone adapter, if necessary), and listen during the call.

If the modems connect, but then you see garbage characters, your host-end modem may not be configured properly. That modem must be configured to talk to the host system at a *locked-in* baud rate, that

matches the baud rate of the host's serial port, regardless of the rate established in the modem-to-modem connection. See section 5.5, *Troubleshooting Modem Connections.*

1.6.2.4   General Problems

If you can't see the characters you're typing, turn off Communicate/Full Duplex

If you see two of every character you type, turn on Communicate/Full Duplex

If you see intelligible characters, but the character positioning is wrong, the host is expecting a different terminal type. Try a different terminal type with Communicate/Terminal Type.

> ➢ If you change the terminal type in Anzio's menu, Anzio will not notify the host that a change has occurred. You may need to save settings, quit Anzio, and restart.

If the screen looks OK, but function keys do not generate the correct response from the host software, you may need to try a different keys file, with Communicate/Read Keys.

If you're having trouble seeing the proper non-Latin (Chinese, Russian, Hebrew, etc.) characters, go to Communicate/Character Sets.

1.7   The Anzio Window

Now let's take another look at Anzio's main window. It conforms to standard Windows user interface guidelines.
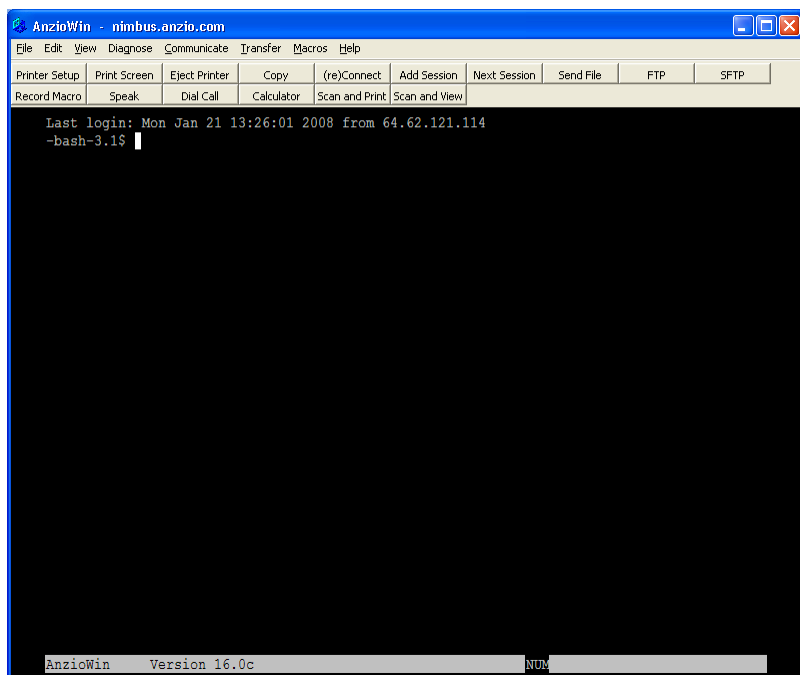
At the top is a Title Bar, with a System menu icon, a Minimize button, a Restore or Zoom button, and a Close button. You can move the window around on your screen by dragging the title bar, or minimize it or quit it with the appropriate buttons. You can also *zoom* it, but we'll get to that in a moment.

Below the Title Bar is the Menu Bar, which shows Anzio's menu options. Below that is the toolbar. At the bottom is the Status Line.

The remainder of the window is the Active Area, which is used by the host program to display characters. This area is divided (although you can't see it) into rows and columns of character cells. To start with, it shows 80 columns and 24 or 25 rows; this is known as the *screen size*. Every character cell is the same size, and can contain a character in a certain *font size*.

> ➢ You can change the number of rows and/or columns with
> View/Screen size. You can change the font size with View/Font size.

As you might imagine, the screen size and the font size are related to the *window size*, the size of the entire space that Anzio occupies. Understanding how Anzio treats that relationship will help you get the best use out of Anzio.

1.7.1   Zoomed Behavior

You can *zoom* or *maximize* Anzio's window by a) clicking the Zoom button on the title bar; b) double-clicking the title bar itself, c) hitting the View/Zoom/Maximize menu item, d) typing ⌷ALT⌷ ⌷Z⌷, or e) right-clicking and selecting Zoom/Maximize from the popup menu. When the window is zoomed, it takes over the entire screen. Anzio will automatically adjust the font size to best display the necessary number of rows and columns in the window space provided.

1.7.2   Unzoomed Behavior

If you repeat the above procedure (or similar), to make the window *restored* or *unzoomed*, Anzio will make the window the necessary size to display the needed rows and columns (screen size) at the needed font size. That is, the window size follows the font size. If you change the font size or the screen size (using the menu), the window size will also follow.

If you *resize* the window, by dragging one of its borders, however, Anzio will automatically adjust the font size to follow the window size. This makes it easy to clear some space on your screen while still seeing all the characters displayed by the host.

1.7.3   80-column and 132-column Modes

Sometimes you will need to display 132 columns of data, such as when previewing a print job at the host system. The host program can send an escape sequence (which varies by terminal type) to switch the "terminal" into 132-column mode. Or, you can change the screen size (rows and/or columns) by using the View/Screen Size menu item. Any of these will cause Anzio to adjust its screen display as follows.

Anzio remembers two sets of preferences; one for 80-column mode (or less), and one for more-than-80-column mode (such as 132). For each mode, Anzio remembers whether the window was zoomed and, if not, what its font size and window position were.

The first time you see it switch to 132-column mode, it is disconcerting. But go ahead and configure it for how you want 132-column mode to look, by either zooming the window or moving and sizing it. After you have set your preferences for both 80-column mode and 132-column mode, you'll be set.

It is also possible for Anzio to remember all these settings between sessions. Just make sure that, in the View menu, you have Track

Window Position checked. When you quit from Anzio, it will ask if you want to save your settings. Answer "Yes".

1.7.4   <u>Other User Preference Items</u>

It is also possible to set various other things that affect how Anzio looks (and sounds). These are generally found in the View menu. You can change screen and font colors, cursor type, beep behavior, and whether or not the status line or gauge line is displayed. You can also set a picture as the *background bitmap*, which is displayed behind the screen text.

More advanced options are available in Edit/Advanced Options, under the General and Screen tabs.

1.8   Exiting from Anzio

To exit Anzio, press the ALT and X keys, or use File/Exit or the EXIT command, or click the Windows Close Box (the "X" in the upper right corner).

If you have changed either the Anzio settings (defaults file) or the defined keys (keys file), you will be prompted to save them. For now, just accept each given file name.

Each time Anzio starts up, it looks for a settings file named AnzioWin.def. If the file is found, Anzio loads those settings. A settings file may include the name of a key file – if so, those defined keys are then read in (see *Loading Key Definitions*, page 51).

You can use multiple settings files, and multiple key files, for different Anzio configurations (for different users or hosts) on the same PC. See *More On Starting Anzio*, page 187.

1.9   Typical Terminal Types

Anzio can emulate many common terminal types. This section describes some of them:

- VT100, VT220, etc.
- *WYSE 50 and WYSE 60*
- *ADDS Viewpoint*
- *SCOANSI*
- *AT386 and AT386-IE*
- *Linux*

- *Terminal Type "Anzio"*

Your host system or application software may support only one particular terminal type. However, if it is flexible, you may be able to choose which terminal type to have Anzio emulate. This section will help you make that choice.

Most terminals support various *character attributes*, which affect the appearance of characters on the screen. Character attributes include reverse video, blinking, underline, intensity, etc. Anzio can emulate these attributes. Some terminal types can also explicitly use colors for foreground (text) and background.

Terminals also vary in the number, names, and placement of special keys, such as function keys, arrows, and "PageUp".

1.9.1   VT100, VT220, etc.

The VT100/220 is the basis of the *ANSI standard terminal*. Each screen position can have its own colors and attributes. The VT320 and VT420 are based on the VT220, with additional features.

The VT100 has only four function keys and a few special keys. The VT220 (and higher) has more function keys, but they do not correspond well to the PC's keyboard.

Anzio in any VTxxx mode responds to control sequences for the VT100, VT102, VT220, VT320, and VT420 (except for some obscure commands). Anzio also provides some of its own control sequences.

Anzio includes several keymap files to handle different requirements for mapping PC keys to VT keys.

1.9.2   WYSE 50 and WYSE 60

The WYSE 50/60 is a widely-supported terminal. The WYSE has many function keys, which have a "standard" definition, as well as several special keys.

The WYSE has an advantage in that its arrow keys send single-byte control codes. Some software seems to prefer this, notably "vi" under UNIX.

Wyse terminals can have local key definitions downloaded from the host. These local definitions, if any, take precedence over any Anzio-defined key with the same name. Downloaded key definitions are not normally saved. The KEYS display (Macros/Display Macros and Keymaps) will

show both definitions, with the local definition on top. See Defining a Key Macro the New Way, page 47.

The WYSE 50 supports both *field* and *screen* attributes. Each field attribute occupies one screen character position. The screen attribute indicates "protected" fields, but does not occupy a screen character. Anzio supports both field and screen attributes, including protected fields.

The WYSE 60 provides character attributes for each screen position, but the attributes do not occupy a position. This approach saves screen real estate, and is preferred for use with Anzio.

➢ Anzio does not support WYSE *page mode*.

1.9.3   ADDS Viewpoint

The ADDS Viewpoint is a commonly-emulated terminal. The original model did not have function keys, so there is little consistency in emulating them. Like the WYSE, arrow keys are single bytes. The ADDS is limited in its attributes, using "tagged" attributes. The host specifies one attribute per screen. Each location on the screen is either tagged or not tagged (one bit). Tagged locations use the given alternate attribute.

1.9.4   SCOANSI

When Anzio is set to SCOANSI, it emulates the console of SCO Open Server UNIX systems. This is a popular choice because it has full explicit color control, as well as many function keys and special keys.

Anzio emulates the console behavior as of Open Server 5.0.6. However, it can be set to a compatibility mode which causes it to behave like 5.0.5 and earlier (in Edit/Advanced Options).

1.9.5   AT386 and AT386-IE

These modes emulates the character-based console of UnixWare. The AT386-IE version supports additional international characters (presumably "ie" stands for "international edition"). These modes support explicit colors and many function keys and special keys.

1.9.6   Linux

This mode emulates the character-based console of Linux. It has explicit color control, and can simultaneously support 16 foreground colors, 16

background colors, blink, and underline. It also has support for many function keys and special keys.

1.9.7   <u>Terminal Type "Anzio"</u>

You can tell your UNIX host that your terminal is really Anzio. The advantages to this are:

- many function keys and special keys are defined
- color is supported
- the host has access to enhanced Anzio capabilities

We have provided some files for use with UNIX and Anzio. Use the following procedure to configure these files:

1. Set Anzio to act as "Anzio", with either Communicate/Terminal Type or the TERM ANZIO command.

2. Load the keys file "AnzioTIC.kys" with File/Read Keys.

3. Upload the *terminfo source file* "Anzio.tic" to your UNIX host, using the *UNIX Simple Upload* procedure on page 87. On the UNIX host, change the name to lower case ("anzio.tic").

4. Tell UNIX to run the *terminfo compiler* "tic" with the "anzio.tic" file. You will need superuser privileges – you may have to ask a local expert for help.

   ```
   tic anzio.tic ⏎
   ```

5. You will probably get warnings, since this file has entries for both UNIX and AIX. Barring fatal errors, your UNIX host should now consider "ANZIO" or "anzio" a legal TERM type, for any program that uses the UNIX terminfo utility. Tell the host to set its TERM variable to "Anzio":

   ```
   TERM=ANZIO; export TERM ⏎
   ```

6. Finally, tell UNIX to initialize the terminal:

   ```
   tput init ⏎
   ```

➢ If the host application software does not use terminfo (such as Word Perfect), it will not understand a TERM of "ANZIO". Rasmussen Software has developed Anzio definitions for popular software – see

the README.TXT file, our web site, or contact us for more information.

1.10   Using Function Keys and Special Keys in Terminal Emulation

To use function keys and special keys (such as the arrows) with your host software, Anzio must be set up to send the correct character sequence for each key. Each terminal type that Anzio can emulate has a basic set of key definitions built in. These can be superseded by definitions contained in a "keys" file. You can either load one of the supplied keys files, or you can define the keys yourself (see Defining a Key Macro the New Way, page 47).

To load a keys file, use File/Read Keys or the READ command. Also, if you change your terminal type with the Communicate/Terminal Type menu item, Anzio will ask whether you want to load the usual keys file for the new terminal type.

Supplied key definition files, such as VT220.kys, are listed in REF _Ref431994059 \w Appendix E, *Distribution Information*, page 235. Each key definition file has a corresponding ".DOC" file explaining it. You can TYPE the ".DOC" file, read it with an editor or viewer, or print it out.

Note that when emulating one of the VT terminals, you may have to experiment with using the various VT keys files. An actual VT terminal has a keyboard that does not match a PC keyboard exactly, especially in the function keys. Anzio's online help has additional information.

Note also that the Backspace key is also a frequent point of confusion. It has its own setting in the menu as Communicate/Backspace.

1.11   Anzio's Mouse Support

Anzio is designed so that mouse actions will do what you would expect them to do, but you can override that. By default (assuming right-hand mouse configuration in Windows):

1. Left-drag will highlight a section of the screen. If you hold the ALT key down when you release the button, the highlighted region is copied to the Windows clipboard. The region is copied in several formats: as OEM text, as ISO text, as Unicode text, as Rich Text (unless disabled), and as a bitmap. In the destination Windows program, use Edit/Paste to paste from the clipboard, or Edit/Paste Special to access the different formats.

2. Left-clicking while holding down the SHIFT key will extend the highlight to the point clicked.

3. Left double-click will send to the host the word that is pointed to by the mouse cursor, followed by a carriage return.

4. Right-click will popup a context menu, offering you several options.

5. Scrolling the wheel up will take the screen into REVIEW (scrollback) mode, where you can see what has scrolled off the top of the screen.

6. If the window is not maximized, scrolling the wheel while holding down the CTRL key will increase or decrease the text size and the window size.

7. If the host system tells Anzio, via escape sequences, that it wants to receive mouse actions, Anzio will comply. Anzio supports several mouse-to-host protocols, including some defined by "xterm" and also the DEC VT "Locator" mode.

You can change the behavior caused by various mouse actions, by going to Edit/Advanced Options/Mouse. There you can also change the shape of the highlight that is created when you drag the mouse.

1.12   Selecting Text Using Keystrokes

In addition to being able to select (highlight) text with the mouse as described above, it is possible to select text with arrow key combinations, *unless* those combinations have been redefined.

> ➢ The highlighted area of the screen is what gets copied to the clipboard, as in other Windows programs. In addition, though, certain commands in Anzio are affected by whether text is highlighted.

All highlighting keystrokes use the SHIFT key in conjunction with a special key. The arrow keys move the highlight in the indicated direction. HOME and END move to the left and right end of the line, respectively.

If you haven't already created a highlight, Anzio will begin by extending the highlight from the current cursor location. Thus the cursor location is one end of the block, and you are moving the other end of the block. You can *switch ends* by hitting the numeric-pad 5 key while holding the SHIFT .

If you do SHIFT PG UP , or you SHIFT ↑ past the top of the screen, Anzio will take you seamlessly into REVIEW mode, with the highlight still in effect.

You can continue to scroll upwards, and thereby select a large part of your REVIEW buffer. To exit REVIEW mode, hit ESC .

2 Help When You Need It

Anzio provides two kinds of online help:

First, a Windows help system is included, and is accessed via the Help menu and the Contents item. You can also get help on specific menu items by hitting ⌨F1 when the menu item is highlighted.

Second, a great deal of information is posted on Rasmussen Software's web site, at http://www.anzio.com.

Oh, and we do actually answer questions by email or telephone!

3   Talking with Anzio

Anzio can act as a simple terminal, just sending and displaying characters. To use all of Anzio's features, though, you will need to enter Anzio commands. Anzio provides menus to access most Anzio commands. The following sections describe:

- *Anzio Menus*
- *Entering Commands*
- *Editing the Command Line*
- *Predefined Function Keys*
- *Using Anzio's Language Support*
- *Using Review Mode to See Previous Data*

3.1   Anzio Menus

Anzio provides a set of menus, with menu items for most Anzio commands.

> ➢ See the online help (Help/Contents) for more information about each menu and item. You can do this by activating the menu, positioning the highlight to the menu item in question, and hitting [F1]

File Menu

- Open input and output files, change directories
- Save settings files
- Read, save, and merge defined key (macro) files
- Set up printers and the Anzio Print Wizard
- Print the screen
- Set and clear the background bitmap
- Exit the Anzio program

Edit Menu

- Copy and paste screen contents, optionally to a file
- Go to Advanced Options

View Menu

- Set user preference items, such as beep, gauge, status line, scrolling, cursor type
- Set screen colors and font

- Set window title, size, and preferred position

Diagnose Menu

- Monitor communications
- Interpret all received characters, including control codes
- Review previous screens of data
- Diagnose problems in establishing communications with the host

Communicate Menu

- Send a BREAK
- Unlock the keyboard
- Set host protocol: linefeed or carriage return, duplex mode, lock mode
- Set terminal emulation parameters: name, backspace character, environment variables, Caps Lock
- Set modem parameters: dial-in number, answerback string
- Hang up the modem

Transfer Menu

- Capture the screen to a file, or to a printer
- Transmit files from the PC to the host, with various protocols
- Receive files from the host to the PC, with various protocols
- Initiate FTP or SFTP interactive file transfer
- Set character and line delay times
- Configure various options for file transfer

Help Menu

- Display on-line help contents and topics
- Show Rasmussen Software support contact information, including the Anzio version and copyright

---

➢ You may customize Anzio using a Windows resource editor to add or remove menu items and accelerator keys. For example, if you remove File/Quit, Alt-X and Alt-x, a user cannot exit a running Anzio.

3.2   Entering Commands

One way to control Anzio is to type in textual commands. Use the menu item Macros/Enter Command, or type [CTRL] [SHIFT] [F] . A "FUNCTION:" prompt will appear on the bottom (status) line. Enter any Anzio command, then press [ENTER] to run the command and return to your "live" screen.

> ➢ Press [ESC] to exit from the FUNCTION prompt without performing any command.

Anzio commands can also be called from:

- A defined key macro – see *Defined Keys and Macros*, page 45.

- The host computer – see *Sending Anzio Commands From The Host*, page 202.

Anzio allows some leeway when entering command names and options:

- Commands may be entered in UPPER or lower case.

- The command name can be abbreviated, to any unique abbreviation. For example, INTERPRET can be called as INT. The only exception is the DEFINE command, which cannot be abbreviated.

- Command options can be shortened, and sometimes omitted. For example, FULL instead of FULL DUP, or DATA 8 instead of DATA BITS 8.

- Spaces are ignored, except as part of defined keys.

- The option ON, such as GAUGE ON, is optional. That is, GAUGE by itself means GAUGE ON.

- Commands having options ON and OFF also accept SWITCH or TOGGLE to change between states. For example, GAUGE SWITCH alternately turns the gauge off and on.

- When a command allows or requires an argument, and you need to enter an argument that contains an embedded space, put double quotes around the argument.

- If you need to embed a double-quote inside a quoted string, put a backslash ("\") before it.

3.3   Editing the Command Line

You can edit a command while entering it, using the following keys:

BACKSPACE

Deletes the character to the left of the cursor, and backs up the cursor.

→ , ←   Moves the cursor right or left.

INS   Switches between *insert* and *overtype* mode. In insert mode, displays INS at the right end of the command line.

DEL   Deletes the character at the cursor.

HOME   Moves the cursor to the beginning of the command.

END   Moves the cursor to the end of the line.

F1   Delete-to-end-of-line – erases all characters from the cursor to the right end of the line.

F7   Used with the KEEP, PRINT, and PICK functions to define a rectangular area (column) on the live screen – see *Data Capture*, page 84.

CTRL P   *Prefix key* used to enter character codes. The next character is included as-is in the command line. For example, to enter an ASCII "escape" character, press CTRL P ESC . Without the prefix, ESC is interpreted by Anzio as an editing command (next).

ESC   Throws away the edited command and returns to the live screen.

➢ To include an ESC in a command, press CTRL P then ESC .

Function keys entered as part of a command are shown in reverse video with the function key name, such as F5. If a shift, control, or alt key is used with the function key, that key is shown by:

> s   SHIFT
> ^ (caret)   CTRL
> a   ALT

Other special keys are displayed by name, or by hex codes.

The line shifts over if you type past the right margin.

3.4   Predefined Function Keys

Anzio, at its core, predefines several function keys for its own use. For instance, F6 initiates an Anzio command. However, any of the function keys may be redefined, either to execute a user macro or to send a control sequence to the host. This is referred to as *overdefining* the key.

➤ Use Macros/Display Macros and Keymaps to see the current function key assignments.

In fact, most function keys are *usually* overdefined. However, their core definitions are listed here because they are sometimes used inside macros, or for other purposes.

If a function key has been overdefined, you can still access its Anzio function using a *modifier key*. For example, if your configuration uses F3 (that is, F3 sends a code sequence to the host), you can access Anzio's core F3 action with SHIFT F3 , ALT F3 , CTRL F3 , or CTRL SHIFT F3 . Of course, you may also overdefine these *modified* function key combinations, but we don't recommend it. See *Overdefining Anzio's Default Function Keys*, page 50.

Some of Anzio's default function keys also have an alternative key combination, as listed below.

F1	"Send defined key". Press F1 , then a defined key – that key is INVOKEd and the results are sent to the host. See *Executing Key Macros*, page 50.
F3	*Function prefix key*, used for emulating the FUNCTION key on certain terminals. Displays "Func:" on the status line and moves the cursor there. The next key you press is sent in ADD128 format: decimal 128 is added to the character code. An exception is if Anzio is emulating an NCR 7900, in which case Anzio sends a hex 02, then the key code, and then an ENTER. Press ENTER alone to exit the Func: prompt without sending anything.
F4 or ALT U	Unlocks the keyboard.
ALT F4	Quits Anzio (if allowed).
F5 or ALT M	HELP key. An old DOS-style help screen is displayed, showing Anzio commands, any open files, and the current status of many options (Baud

rate, duplex mode, etc.). Anzio commands can be typed in at the bottom of the HELP screen.

[F6] or [CTRL] [SHIFT] [F] "Anzio function" – displays the FUNCTION prompt for you to enter an Anzio command. [F6] is also the initiator of an embedded command – a command inside a macro.

[F9] or [ALT] [A] PANIC BUTTON (Grand abort)– aborts a defined key in progress, turns TRANSMIT OFF, exits any endless loops, and empties the keyboard buffer.

[F10] or [ALT] [B] BREAK key – sends the BREAK signal to the host computer. Can also be pressed at any time to clear the type-ahead buffer and unlock the keyboard.

[CTRL] TAB Switches focus to the next instance of Anzio running on the PC.

[CTRL] [SHIFT] TAB Switches focus to the previous instance of Anzio running on the PC.

[CTRL] Backspace Word delete left – deletes characters to the left until start of word or start of field.

[CTRL] [C] Copies part or all of the screen to the clipboard, OR sends the ctrl-C code to the host, as configured in Edit/Advanced Options/Keys.

[CTRL] [V] Pastes text from the clipboard, OR sends the ctrl-V code to the host, as configured in Edit/Advanced Options/Keys.

3.5   Using Anzio's Language Support

Anzio is able to handle many kinds of characters, including Latin, Cyrillic, Hebrew, Arabic, Turkish, Chinese, Japanese, Korean, and more, as well as line-drawing characters and many technical symbols. Internally, Anzio stores all these characters in Unicode, a 16-bit coding scheme which defines all these characters. Anzio uses sophisticated techniques, working with routines provided by Windows, to display, print, and accept from the keyboard these characters.

But the host system that Anzio is connected to may not work in Unicode; even if it does, passing 16-bit characters is a problem. So when communicating with the host, Anzio translates characters between Unicode and some *encoding scheme*, also known as a *character set*.

Anzio supports many different character sets. All are based on ASCII, where, for instance, hex 41 is always "A". However, the host system may need to store, display, print, and accept as input a character such as "â". There are various standards for coding these non-ASCII characters, and Anzio supports many of them.

For older systems, where it was important to keep characters limited to 7 bits, coding schemes were developed that replaced less used special characters, such as "\", with the accented characters used in, for instance, Norway. These coding schemes are called National Replacement Character sets, or NRCs.

More recent efforts left all the 7-bit codes the same as ASCII, but established standards for 8-bit codes, that is, codes from hex 80 to hex FF (decimal 128 – 255). These are called Single Byte Character Sets (SBCS).

Also, schemes have been developed to store characters in languages that have more than 256 characters, such as Chinese, Japanese, and Korean, using more than one byte per character. These are called Double Byte Character Sets (DBCS) or Multi Byte Character Sets (MBCS). They take one byte for some characters (including ASCII) and two bytes for others.

Finally, Unicode is often represented as a sequence of one to four bytes per character. This format is called UTF-8.

Anzio is able to support NRCs, and also the following character sets:

- ISO - the "Windows" character set, which varies based on the language of your Windows installation.
- DEC – very similar to ISO 8859-1.
- OEM – sometimes called "linedraw" or "DOS". Corresponds to the DOS codepage on your Windows installation.
- ISO-8859-x, where "x" is 2 to 9 or 15. These are used in Europe and elsewhere.
- ASMO449 – Arabic
- KOI8-R – Russian
- VISCII – Vietnamese
- DOS and Windows code pages, including Far East code pages. You must have installed Windows support for the particular codepage.
- Library of Congress USMARC

- Byte-encoded Unicode (UFT-8)
- CCCII for Chinese, Japanese, and Korean characters.
- T160E encodings of diacritics.
- Custom encodings, as defined by the user

> ➢ To use these encodings, you must set the terminal emulation type to VT100, VT220, VT320, VT420, SCOANSI, LINUX, AT386, or ANZIO.

There are four areas affected by character encodings:

- Characters received from the host must be displayed properly.
- Characters typed on the keyboard, or pasted from the clipboard, may need to be translated before being sent to the host.
- Characters sent from the host as part of a passthrough print job need to be printed properly.
- Characters received from the host and CAPTUREd to a file must be encoded properly.

3.5.1   Setting the Host's Character Set

Click the Communicate/Character Set menu item. This will bring up a dialog box that allows you to choose a "Character set for screen". The "7-bit" selection refers to NRCs, as explained above. If you do NOT need an NRC, as most people won't, select "International".

Under "8-bit", specify one of the sets as described above. If in doubt, choose "ISO".

You can also use the CHARSET command to specify the host's character encoding.

3.5.2   Setting Print Encodings

In order to passthrough-print non-ASCII characters, you must use AnzioWin (not Anzio Lite), and turn on Print Wizard. With any other configuration, coding is determined partly by the terminal type you are emulating, and partly by the printer itself. However, screen prints should still work.

To tell AnzioWin what coding scheme to apply to passthrough print, go to Communicate/Character Set, and make settings in the area labeled "Character set for passthrough print", similar to what was described above.

The printer's encoding can also be set with the PASSTHRU-CP command.

3.5.3   Keystroke Entry

You may need to enter any of a wide range of characters into Anzio, or through Anzio into the host system. Some of these characters, or character-diacritic combinations, may not exist on your keyboard. This section describes the many ways of entering characters into Anzio.

3.5.3.1   Windows Keyboard Switching

Your Windows system can be configured to support various keyboards. For instance, you can have both an English (US) and a Russian keyboard driver installed on your Windows, and then you can switch between them with a special key combination. See your Windows documentation for more information. Anzio will correctly understand keystrokes entered from any Windows keyboard driver.

Note that some versions of Windows have an "on-screen keyboard" available under the "accessibility" section. This feature may help you know what characters are assigned to what keys.

3.5.3.2   Alt-numeric Entry

Since the days of DOS, it has been possible to enter non-ASCII characters using the Alt key and the numeric pad (NOT the numbers at the top of the alpha keyboard). For instance, alt-144 enters an E-acute (on a US Windows system). This number is taken from the "DOS" or "OEM" character set.

For a character from the Windows ("ANSI" or "ISO") set, precede the number with a zero, such as Alt-0255 for a y-umlaut.

In Anzio, you can also enter any Unicode character by its *decimal* value. For instance, a Y-tilde has a Unicode value of hex 1EF8, which is decimal 7928. So you can do Alt-7928 to enter this character.

Finally, you can enter a Unicode character by its *hex* value, by starting with an Alt-plus. In this one case, you will need to use characters **not** on the numeric pad, for A through F. So for the example above, hold the Alt key and type +1EF8 (where "+", "1", and "8" are on the numeric pad).

➢ To see charts of Unicode characters and their numeric values, go to www.unicode.org.

3.5.3.3 Dead-key Entry

Anzio has built-in support for treating certain control-key combinations as dead keys, that will combine with the next character. This is modeled after Microsoft Word. The following table shows each key combination and its meaning. Note that in some cases you must include SHIFT in order to get the correct character:

Key combination	Meaning	Combines with (examples)
Ctrl-`	grave accent	a e i o u A E I O U
Ctrl-'	acute accent	a e i o u A E I O U d D
Ctrl-6	circumflex	a e i o u A E I O U
Ctrl-:	umlaut	a e i o u A E I O U y space
Ctrl-,	cedilla	c C space
Ctrl-~	tilde	a n o A N O
Ctrl-&	ligature	a s A
Ctrl-/	slash	o O
Ctrl-.	dot below	a b d e A B D E
Ctrl-?	hook above	a e A E
Ctrl-(	breve	a e A E

So for example, if you type ctrl-colon (which is ctrl-shift-semicolon), you will see nothing; then when you type "a", Anzio will get an a-umlaut (ä).

> ➢ There are many more combinations available than are shown here. Any combination defined in Unicode, which involves one of the diacritics above, should be supported. However, you may be limited in what your host system will accept, or you may be limited by the character set you are using.

3.5.3.4 Input Method Editors (IMEs)

If you have a Far East version of Windows, such as Korean, your keyboard driver for Korean will be an Input Method Editor (IME). When the IME is active, you go through a several steps to choose each character. Windows 2000 and later have support for all languages that Windows supports. So even if you have an English version of Windows 2000, you can enable support for Chinese, Japanese, and Korean, and obtain their IMEs. Anzio will properly recognize characters coming from all these IMEs.

3.5.3.5   Add-on Programs for Far East Input

Anzio will also work with most add-on programs for Far East input, such as Union Way, WinMass, Twin Bridge, etc. Because Anzio can handle all the text display and printing tasks, these programs are useful only for providing input methods.

It is important to coordinate the encoding scheme that an add-on uses to output a keystroke, with what Anzio expects as input. See *Editing the DEF File* page 221.

3.5.3.6   COMPOSE and UNICODE Commands

Anzio's command language includes two commands to aid in character input.

The COMPOSE command is useful to combine a character and a diacritic.

The UNICODE command translates a hexadecimal value to its Unicode character equivalent.

As with other Anzio commands, these can be programmed into macros and used in a variety of ways.

3.5.3.7   Paste from Clipboard

Whenever you paste into Anzio, whether to the host program or to the Function prompt, Anzio will pull Unicode characters from the clipboard if possible. So you can Copy data to the clipboard from various multilingual applications, such as Internet Explorer, Word, or Outlook, and paste it into Anzio. The converse is also true; you can copy from Anzio and paste into other applications.

3.5.3.8   Unicode Characters in Macros

Anzio's macro ("defined key") feature can handle any Unicode character. So it is possible to define macros that send one or more characters to the host system, with proper encoding.

3.5.4   <u>Automatic Font Selection</u>

Anzio's ability to display and print Chinese, Russian, Turkish, and more implies a need to deal with multiple fonts. In fact, Anzio has a unique ability to find appropriate fonts from those installed in Windows.

You configure a primary font; for the screen (in View/Screen font), and for the printer (in File/Printer Font). As Anzio is displaying (or printing), if

it encounters any non-ASCII character, it checks to make sure that character is included in the primary font. If it is not, Anzio will search through the fonts installed on Windows to find a font that contains the character, and use that font for this character. This operation is automatic. So you can go from displaying English to Chinese to Russian to Turkish, with no user intervention. Preference is given to mono-spaced fonts, and to fonts that have the correct width for Far East characters.

If Anzio is unable to find a character in ANY font installed in Windows, it will display that character as a rectangle with an X in it.

You can download and install non-Roman fonts, such as for Far East languages, from Microsoft's web site. To do so, simply point your Internet Explorer to the Internet Explorer update web site at Microsoft, and install the particular language support (such as Korean). This will get you the fonts you need, and they will be available to Anzio the next time you run it. You may also be able to install the fonts through your Windows Control Panel, depending on your version of Windows.

Because diacritic handling is important to many of our customers who are libraries, Anzio also has special handling of character/diacritic combinations. It can handle combinations that are not part of the Unicode standard, by combining them dynamically. When displaying combinations, Anzio goes to considerable lengths to make sure the diacritics are available and properly placed, to always render good results.

3.5.5   Changing the Language of the Menus

You can change the language used in Anzio's menus, messages, and dialog boxes, from the native English to Spanish, French, etc. Go to View/Menu Language, or use the MENULANGUAGE command.

Menu language support is provided in the form of files named ANZIO*xxx*.DLL, where *xxx* is an abbreviation for the language name. At the time of this writing, we provide Spanish, Italian, and French in addition to English. If you are interested in creating another translation, please contact us.

3.6   Using Review Mode to See Previous Data

Anzio saves incoming data in a special *review memory*, large enough to contain many screens full of data. The review memory is initially blank (filled with spaces). The amount of space in the review memory is configurable; see page 209.

> ➤ To start REVIEW mode, press ⌊ALT⌋ ⌊R⌋ , or use Diagnose/Review or the REVIEW command. Exit REVIEW mode with 🔲 or ⌊ESC⌋ . The mouse wheel can also take you in and out of Review mode, depending on its configuration.

When REVIEW mode is active, the status line displays

```
USE UP-ARROW, DOWN-ARROW TO REVIEW SCREEN
```

Anzio stores in review memory all lines that are scrolled off the top of the screen. In addition, any screens "erased" by the host will be "scrolled" into memory instead.

3.6.1   <u>REVIEW Mode Scroll Keys</u>

REVIEW mode uses only the following keys:

🔲 or ⌊ESC⌋	Restores the working screen and exits REVIEW mode.
⌊↑⌋ , ⌊↓⌋	Moves the view window up or down one line.
⌊PG UP⌋ , ⌊PG DN⌋	Moves the view window up or down one screenful.
⌊←⌋ , ⌊→⌋	Moves the view window left or right one column, if possible.
⌊HOME⌋ , ⌊END⌋	Moves the view window to the far left or far right, if possible.
⌊P⌋	Prints the current screen.

Also in REVIEW mode, you can highlight text with the mouse or with key combinations as explained above.

4   Defined Keys and Macros

Anzio lets you assign characters, strings, and commands (macros) to almost any keyboard key. When you press a *defined key*, Anzio looks up the key definition and processes that definition as if you just typed it in. This feature is used to establish what sequences special keys send out, as well as to store commands into macros for various purposes.

This capability covers, in a unified approach, two kinds of key definition. First, it covers terminal-type special keys. That is, if Anzio is emulating a VT220, and you hit the F10 key, what character sequence should be sent to the server? If the Anzio program doesn't contain the key definition that your host system needs, you may have to *define* the F10 key.

Secondly, you can use defined keys as a shortcut method of entering any common keystroke sequence to the host. For example, if you frequently have to type the host command "DEALLOCATE", you could assign it to the F10 key by clicking on Macros/Enter command, then typing

        DEFINE F10 DEALLOCATE⏎

The next time you need to enter that command, just press F10 , and Anzio will send the characters as defined.

> ➢ There are other ways to enter macros also; this is just one example.

You may also include most Anzio functions in a defined key, as described below. While a defined key is running, it can display strings to the user (*prompts*) that are not sent to the host, and can wait for the user to enter strings or select filenames.

After defining one or more keys, you can save the current set of key definitions to a file. You can read in or merge other key definition files, respectively replacing or adding to the current set.

Use the Macros/Display macros and keymaps menu item to see the current defined key sequences. This display shows all system-assigned keys and user-defined keys. Each defined key is displayed in reverse half video (or the color equivalent), followed by its contents. Non-display characters are shown in reverse video by name or as an ASCII hex number (01 for a *control-A, etc.*). You can also display (and edit) individual key definitions using Macros/Edit.

The following sections describe:

- *Which Keys Can I Define?*
- *Defining a Key Macro the New Way*
- *Editing a Key Definition*
- *Executing Key Macros*
- *Overdefining Anzio's Default Function Keys*
- *Using Key Definition Files*
- *Special Characters Inside Defined Keys*
- *Using Anzio Functions in a Defined Key*
- *Nesting Defined Keys*
- *Special Uses of Macros*

4.1   Which Keys Can I Define?

There are four general categories of keys that can be DEFINEd.

- All printable ASCII characters, from space (hex 20) to tilde (hex 7E). Definitions for printable keys are NOT triggered by simply pressing that key; other steps are required as described below.
- Special keyboard keys, including function keys, named keys such as [HOME] and the numeric pad period [.]. The numeric pad keys are only accessible with [NUM LOCK] off, except that separate definitions are possible for numeric pad [/] [*] [-] [+] [ENTER] with [NUM LOCK] on.
- ASCII control codes, hex 00 through hex 1F.
- Alt-key combinations.

Function keys [F1] through [F10] can be defined four ways each: normal, with [SHIFT] , with [CTRL] , and with [ALT] . Also, if your system supports keys F11 and F12, Anzio will support them also, in all four states.

> ➤ Anzio treats the [CTRL] [SHIFT] combination as synonymous with [ALT] .
> So for instance, if you define [CTRL] [SHIFT] [F1] , it will be recorded as [ALT] [F1] , and you can press either key combination to invoke it.

The [NUM LOCK] key can be defined if necessary. Most often, this is done in order to make the top four keys on the numeric pad emulate keys PF1 through PF4 of a VT terminal, for users who are accustomed to that. To allow the [NUM LOCK] key to be programmed, go to Edit/Advanced options (see page 211). To load standard PF1 to PF4 key definitions, go to File/Merge keys, and select the file "VT-NUM.KYS". Note that setting up Anzio in this configuration often leads to user confusion over the operation of the [NUM LOCK] key.

> ➢ It is possible to define a keystroke combination that is normally used as a *menu accelerator* key, such as ALT Z which normally does a Zoom. Macros take precedence over accelerator keys.

4.2   Defining a Key Macro the New Way

Anzio's older, text-based method of creating macros is described below. It is still an option, but the new methods are easier.

4.2.1   Recording a Macro

The easiest way to create a macro is to have Anzio *record* it while you type the characters that go into the macro. Do as follows:

1. Go to the Macros menu and click Record. This starts the recording session.

2. Type the keystrokes that the macro should send to the host. These can be alphabetic keys, numeric keys, arrow keys, function keys, etc.

3. If needed, perform menu operations. For instance, go to File/Printer Setup, and configure a set of printing parameters. Anzio will add to the recording the *command equivalents* of each operation.

4. Go back to the Macros menu, and click Record again to turn it off.

5. You will be prompted (at the bottom of the screen) for a key to attach your macro to. Hit the appropriate key.

6. If a macro is already attached to that key, you will have a chance to verify your selection.

7. If you would like to see or edit the macro, go to Macros/Edit, and proceeed as explained in the next section.

4.2.2   Using the Macro Edit Dialog

Go in the menu system to Macros/Edit, to see the following dialog:
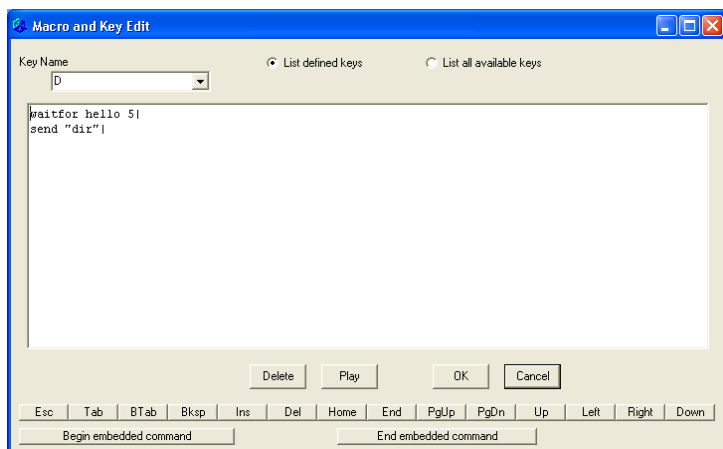
Here you can view, create, edit, delete, and execute ("play") macros.

Select or enter the key to work on in the "Key Name" slot. If there is an existing definition, it will appear in the main box, where you can edit it. If a macro does not exist for an entered key, you can type one in.

➢ It's generally easier to *record* a macro than to type one in. See section above.

If you display an old-style macro, that was created with the DEFINE process described below, Anzio will update the macro to the new style. For instance, keystrokes for the host, but not included in a SEND or SEND/N command, will be placed inside a SEND/N.

Here are some general rules and pointers for creating or editing a macro:

- Each line should start with a command word. Commands are detailed in the reference section later in this manual.
- Each command should end with a pipe symbol ("|").
- For keystrokes to be sent to the host, use the SEND or SEND/N command.
- To enter a special key inside a macro, use one of the buttons, such as "PgDn".
- To embed one command inside another command, use the buttons "Begin embedded command" and "End embedded command". These will show as double-less-than and double-greater-than, respectively. Type the embedded command between them.

- ▪ Some characters have special meanings in macros. See Special Characters Inside Defined Keys, page 52.

4.3 Defining a Key Macro the Old Way

It is still possible to define and manipulate key definitions the old, command-oriented way, as follows.

> ➤ For a newer approach, see above.

Key macros are defined with the DEFINE command. Click on Macros/Enter command, then type:

> DEFINE *x string*⏎

where *x* is the key to be defined, and *string* will become its contents. There must be one space between DEFINE and the key *x*, and another space between the key and the definition *string*.

When the macro *x* is executed, characters in *string* are, by default, sent to the host system, unless they fall into one of the special cases below.

> ➤ If you redefine an existing defined key, the previous definition is de-leted without a warning.

4.3.1 Undefining a Key

You can define a key with a null definition, to override a previous key definition. Enter nothing for *string*:

> DEFINE *x* ⏎

If you later execute key *x*, the default code for that key will be sent to the host. You can also *block* a key, so that it sends nothing.

4.3.2 Blocking a Key

You may need to block some keys, so that an Anzio user can't accidentally send those character codes to the host. For example, ↑ , ↓ , ESC , CTRL C , and so forth.

To block a key, define it with "do-nothing" empty braces {}:

> DEFINE CTRL C {}⏎

If you later press CTRL C , nothing happens – the CTRL C keycode will not be sent to the host.

The empty braces are a special case of a "user prompt" – see *Displaying Text to the User*, page 53.

4.4   Editing a Key Definition

While you are entering a defined key, you can use command-line editing, as described in *Editing the Command Line*, page 35.

To return to an existing key definition and make changes, use DEFINE with a question mark:

```
DEFINE x?
```

When Anzio sees the question mark after the key (without a space), it displays the existing contents of the key. You can then edit the definition. Press ⏎ to finish editing.

4.4.1   Copying a Defined Key

To copy a definition from one key to another, begin editing the existing key, as just described, then back up and change the key name. For instance, to copy key X's definition to key Y, enter:

```
DEFINE X?
```

When Anzio displays the current definition of X, back up and replace X with Y, then press ⏎.

4.5   Executing Key Macros

To execute a key macro defined for a *regular printable* key, press some form of F1 then the key, for example, ALT F1 then X. This assumes ALT F1 has not been redefined.

To execute a key macro defined for a function key or control character, just press the key(s), for example F10 or CTRL X.

You can also execute a macro with the Macros/Run macro menu command. Just click this menu item, then at the "Transmit Key" prompt, enter the key.

4.6   Overdefining Anzio's Default Function Keys

Anzio's predefined function keys, such as F5 for Help and F6 for Function, and its menu accelerator keys can be *overdefined* with your own definition:

```
DEFINE F5 . . .
```

In fact, for most terminal types, these function keys WILL be overdefined. For this reason, Anzio predefines all variations of its function keys as well: ALT F5 , CTRL F5 , and SHIFT F5 will all display the Help screen unless overdefined.

> ➤ We strongly advise you not to redefine ALT A , CTRL SHIFT F , ALT M , ALT U , or ALT X .

4.7   Using Key Definition Files

When Anzio starts up, it can read a *key definition file*. The key file's name is contained in the settings file.

If you want Anzio to start up without reading any key file or settings file, use the command line parameter NONE – see *'NONE' For Clean Startup*, page 191.

Anzio remembers the name of the last key file read in, either during startup or later. This key file name can be saved in Anzio's settings file for use the next time.

> ➤ If you change a key definition, then exit from Anzio, Anzio will ask if you want to save the changes to the default key file.

4.7.1   Saving Key Definitions

To save the current set of defined keys in the current key file, do File/Save Keys, or use the SAVE command:

        SAVE ⏎

The SAVE command without a filename saves the current key definitions to the default key file.

To save the key definitions to a different file, do File/Save Keys As, or enter the command:

        SAVE *filename* ⏎

4.7.2   Loading Key Definitions

There are two ways to load key definitions from a key file: *read* and *merge*:

- When you **read** in a key file, all existing definitions are erased before loading the new definitions. System defined keys will still be used unless overridden by new definitions.

- When you **merge** in a key file, the new definitions are added to the existing ones. A merge will overwrite an existing definition with the new one only if they both have the same key.

To read in a key file, use File/Read Keys or the READ command:

```
READ filename ⏎
```

> ➢ The keys file can also be read from a web or FTP server. Simply enter the URL of the file as the *filename*.

To merge in a key file, use File/Merge Keys or the MERGE command:

```
MERGE filename ⏎
```

4.8   Special Characters Inside Defined Keys

Some characters and keys have special meaning in a defined key:

- Tabs – see *Tabs*
- | vertical bar – see *Using '|' For ⏎*
- # number sign – see *Waiting for User Input*
- {} braces – see *Displaying Text to the User*
- ~ tilde – see *Quoting Special Keys With '~'*
- F6 call function –see *Using Anzio Functions in a Defined Key*

4.8.1   Tabs

Tab characters may be included in a defined key by simply pressing the Tab ⭾ key. A tab is displayed as a reverse video "TAB", but is just one keystroke in the defined key. When the defined key is sent, the tab is processed according to the current tab settings, just as though you had typed it.

4.8.2   Using '|' For ⏎

When you are entering a command from the keyboard, a carriage return ⏎ is required to end the command and begin processing. In a defined key, use the vertical bar character '|' to represent an end-of-command ⏎.

For example, to define key F10 to tell Unix/Linux to list the current directory, enter the following:

```
DEFINE F10  ls|
```

After this definition, press DEFINE ⌐F10⌐ and Anzio will send to the host:

      ls⌐

4.8.3   <u>Waiting for User Input</u>

In a defined key, the number sign '#' causes Anzio to stop for operator input. When this character is found while processing a macro, the Anzio program will stop transmission, accept your entry up to a ⌐ and send the entry (the ⌐ is not sent).

Transmission of the key string continues with the character following the '#'. For example, to create a key sequence to copy files on a Unix system to the /tmp directory, enter:

      DEFINE ⌐F10⌐ cp # /tmp|

Then, when you enter ⌐F10⌐ , the PC will send to the host:

      cp

Then it will wait for you to enter text (a file name). Any key you enter will be sent, until (but not including) the ⌐. The "defined key" then takes over, and sends:

     /tmp⌐

> ➢ There are other more sophisticated ways to prompt the user for input. See the PROMPT/S command.

4.8.4   <u>Displaying Text to the User</u>

To display, but not send, text in a defined string, enclose it in {curly braces}. When a left brace { is encountered during transmission, all characters up to the right brace } are displayed on the PC, but not sent to the host. The special defined key characters '|' and '#' are not processed inside curly braces, just displayed on the screen.

A typical use of displayed text is to place prompts on the screen for use with user input (#), described above.

> ➢ You may include escape sequences, such as for cursor movement, in locally displayed text.

Here is a UNIX example, defining the ⌐F10⌐ key to call the vi editor with a user-entered filename:

      DEFINE ⌐F10⌐ vi {Enter file to edit}# | ⌐

When you enter ⒡⒑ , Anzio sends:

> vi ⎵SPACE⎵

and then displays on the PC screen:

> Enter file to edit

After you enter a file name and press ⏎, Anzio sends the filename and a ⏎ (for the | at the end of the key definition).

> ➢ There are more sophisticated ways to display information to the user. See the MESSAGE command.

4.8.5   Quoting Special Keys With '~'

Use a tilde ~ to "quote" a special character. The character after a tilde is transmitted, not interpreted. So, to program a key to send {A}, enter:

> DEFINE a ~{A} ⏎

The tilde causes the left brace to be sent, rather than displaying "A" as a user prompt. You don't need to quote the right brace, because Anzio did not "see" the starting left brace and so is not expecting an ending brace.

4.8.6   Non-ASCII Characters

Defined keys can contain any Unicode character, including combining diacritics, Cyrillic, Chinese, etc. These will be encoded properly, according to the host's Character Set, when they are transmitted to the host. Characters can be pasted in from the clipboard, or can by entered using the numeric pad as described in *Alt-numeric Entry*, page 40

4.9   Using Anzio Functions in a Defined Key

Inside a defined key, you can call most Anzio functions with ⒡⒍ *functionName*.

For instance, if every day you had to open a file named UPFILE and send it to the host, you could enter:

> DEFINE S ⒡⒍OPENI UPFILE | ⒡⒍TRANSMIT | ⏎

> ➢ Each embedded ⒡⒍ function must end with a vertical bar |, as shown, to represent the end-of-function ⏎.

> ➤ F6 is recognized inside a defined key, even if that function key is overdefined – see *Overdefining Anzio's Default Function Keys*, page 50.

You can also use the "wait for user entry" special character # in an embedded function inside a defined key. For example, if you wanted to enter a filename (instead of UPFILE, above):

> DEFINE S F6 OPENI # | F6 TRANSMIT | ⏎

4.9.1   Keystrokes For Another Function

The SEND command reads characters from a screen line and returns those characters to a function or to the host. For example, to open for input a file whose name is in the first 12 characters of screen line 17:

> DEFINE O F6 OPENI F6 SEND 1 12 17 17||⏎

Several of Anzio's functions can also take keystrokes from a defined key. For instance:

> DEFINE PG UP   F6 REVIEW| PG UP

defines the PG UP key to start REVIEW mode ("F6 REVIEW|"). The first key that REVIEW sees is its command key PG UP , telling REVIEW to show the prior screenful.

4.9.2   File and Screen Selection Functions

Anzio provides two selection functions for use inside a defined key, DIR/S and MENUBAR:

DIR/S	Displays a directory browsing dialog for the user to select a file. Normally, DIR/S sends its result to the host (indicated by "/S"), but inside a defined key, the result is passed back as user input.
MENUBAR	Sets up a selection grid on the screen, with one or more columns. The user selects a grid location, and the characters on the screen at those positions are passed back as user input.
	For example, if the host has displayed a list of file names, or a table of data, MENUBAR lets the user select an entry right off the screen.

4.9.2.1 DIR/S Example

If you need to transmit a particular type of file frequently, you may define a macro to first select a file, then transmit it:

```
DEFINE O OPENI F6 DIR/S *.DAT|#||
          F6 TRANSMIT | ⏎
```

In this case, the filename OPENI is expecting comes from the DIR/S command. DIR/S presents a directory of all files matching "*.DAT". When you select a file, DIR/S returns the name to OPENI. Next, TRANSMIT sends the opened file.

Note the characters after the DIR/S command: The first vertical bar is the end of the F6 DIR/S command, and the # tells Anzio to wait for operator input.   The second vertical bar is the end of the # command, and the last vertical bar is the end of the OPENI command.

4.9.2.2 MENUBAR Example

Suppose that the host has displayed a list of filenames on the screen, and you want to receive one of these files. If you know that the host will display the filenames in columns 1-13, you can call MENUBAR in your key definition:

```
DEFINE K F6 KRECEIVE
          F6 MENUBAR 1 13 1 23 0 0 1 13|#||
          ⏎
```

The KRECEIVE command's input filename comes from the MENUBAR function. The numbers are 1 13 for the start and end column, 1 23 for the start and end lines, 0 0 for no second column, and 1 13 for the start and end returned character offsets.

Note the characters after the MENUBAR command: the first vertical bar is the end of the F6 MENUBAR command, and the # tells Anzio to wait for operator input.   The second vertical bar is the end of the # command, and the last vertical bar is the end of the KRECEIVE command.

4.10   Nesting Defined Keys

You can use the CALL and INVOKE commands inside a defined key to start another defined key. CALL X starts defined key X, and returns to the original defined key when X is done.

INVOKE X starts defined key X, abandoning the original defined key. When X is done, control returns to the keyboard.

4.11   Special Uses of Macros

There are some special ways macros (defined keys) can be used.

DEFINE the macro as described above. For simplicity in this section, we will assume that these macros are attached to alphanumeric keys.

Remember that macros get saved in the keys (.KYS) file, so be sure to save keys.

4.11.1   Startup Macro

Anzio can run a designated macro at startup, in order to automate the process of logging in and doing additional commands.

> ➢ When a startup macro is used, the Login Wizard is not invoked.

You can specify a particular macro to always run on startup, by configuring that in the settings file. To do this, go to Edit/Advanced Options/Keys, click on Startup macro, and enter the letter of the macro in the box. Then quit, saving settings. The next time Anzio is run, the macro will begin executing as soon as the connection with the host is established.

Alternatively, you can specify a start macro in the command line that invokes Anzio, whether that is typed manually, initiated by a start menu item, or initiated by a desktop icon. Simply include the letter of the macro as a command line parameter. See *Single Character – Start Defined Key*, page 191.

A typical startup macro would do a WAITFOR for a login prompt, send the username, WAITFOR the password prompt, send the password, WAITFOR a particular prompt, send a command, etc.

4.11.2   Event Macros

You can attach macros to several Anzio "events". That is, when something happens in Anzio, a certain macro will be initiated. In fact, the startup macro described just above is one example of this.

Other event macros can be specified with the EVENT command, and are described in the explanation of EVENT in the reference section. Examples include ONQUIT (triggered when the user tries to quit), ONBEEP (triggered when the host sends a bell character, and so forth.

4.11.3   <u>Onerror Macro</u>

You may need to have some error handling in your macro processing. For instance, if Anzio is running a macro that opens an input file, but that file does not exist, you may want Anzio to handle that cleanly. For this purpose there is the ONERROR macro.

You define this macro with the DEFINE command, as with any other macro. Let's assume it's the "E" macro. Then use the ONERROR command to designate it as the ONERROR macro:

```
ONERROR E
```

Of course, this ONERROR command could be part of another macro. Now start your main macro. At any point that a command fails, ALL nested executing macros will be terminated, and the "E" macro will be invoked. The ONERROR macro remains in effect until it is cancelled or replaced.

Some examples of commands that can trigger ONERROR:

WAITFOR (with timeout)	Times out
MKDIR	Illegal directory name
CD	Directory not found
LAUNCH	Can't start program
Various OPENs	File not found, etc.
COPY	File not found
RENAME	File not found, target exists

4.11.4   <u>Code Activated Switch Macros</u>

Anzio's passthrough print can be configured so that receipt of certain control codes, in the print data stream, triggers certain actions. More specifically, a particular control character would trigger a particular macro, which might flush a print job, switch to a different printer, and begin printing again on the new printer.

A range of 6 macros is associated with 6 control characters, control-A through control-F. The PRINTMACRO command specifies what the *first* of these 6 macros is; that is, which one is associated with control-A. For instance, if you did:

```
PRINTMACRO a
```

then receipt of a control-A would trigger the "a" macro, control-B would trigger the "b" macro, etc., up to "f".

When Anzio's passthrough print routine detects a triggering control character, it quits from passthrough print and then CALLs the appropriate macro.

5   Using A Modem to Call the Host

You may need to connect to your host system using a modem. Anzio uses Windows drivers to talk to modems, and should be able to work with any modem that your Windows system supports.

> ➢ Note: The old technique of talking to the modem as a serial device is no longer recommended.

The following sections describe:
- *Types of Modem Connections*
- *Making the Modem Connection*
- *Signing Off*
- *Modem Setup*
- *Troubleshooting Modem Connections*

5.1   Types of Modem Connections

It is important to understand the type of modem connection you need. After reading the next two subsections, if you are still unsure, check with whoever supports your host system.

5.1.1   Direct Dial Modems

This is the "old school" of modem connections. A modem on your PC dials a modem on the host system. Once that modem answers, you are connected to the host computer system via a serial connection, and typically you get a login prompt. If this is the kind of connection you need, this is the right part of this manual.

5.1.2   Network Modem Connections

In this approach, the networking software on your PC dials a modem either a) connected to your host system's network, or b) at an Internet Service Provider (ISP). This establishes a network-over-modem connection, and your PC becomes part of the network at the other end, possibly part of the Internet. Once connected, you can run a variety of network protocols over the modem connection (including a telnet connection). The term "PPP" applies to this kind of connection.

> ➢ If this describes the connection you need, then this chapter is NOT relevant. Instead, you would want to establish the connection

BEFORE you run Anzio, and then tell Anzio to connect via a network protocol, such as telnet, rlogin, or SSH.

5.2 Making the Modem Connection

In Anzio's Communicate/Setup dialog, choose "Modem" as your type of connection. If more than one modem is listed, select the one you wish to use. Enter the phone number to be dialed. Click "OK". Anzio should dial the modem, establish a connection with the host-end modem, and present a login prompt as generated by the host system.

5.3 Signing Off

Signing off is the process of disconnecting the modems, and "hanging up" the phone line. Either modem can initiate the process, depending on the host operating system. When one modem hangs up, the other will detect that and hang itself up, notifying the computer to which it is connected.

To end the session from the host, enter a host command such as "logout", "exit", or "BYE". On most operating systems, the host then tells its modem to hang up. When *your* modem detects the hangup ("carrier drop"), it hangs up, and then Anzio behaves as indicated in the Communicate/On Line Drop setting. Anzio can quit, redial, or stay running in a "not connected" state.

To hang up the modem from your end, use the Communicate/Hangup menu item or the HANGUP command.

5.4 Modem Setup

If it is necessary, you can exercise more control over the modem on your PC. In the Communication Settings dialog, click the Setup button next to the list of modems. This brings up a Windows standard "Modem Properties" dialog. Following are some notes on items you might want or need to change.

5.4.1 Modem Connection Speed

Ordinarily, the two modems used in the connection will negotiate a maximum possible speed, for the greatest throughput. However, you may need to force a lower speed. You can **try** to do this by setting "Maximum speed" to a lower value. However, note that many modems, especially so-called Winmodems, do not obey this setting. You can

determine what connection speed is actually established by looking at the diagnostics produced by Diagnose/Debug communication.

5.4.2   Data Bits and Parity

In the Connection tab, you can set the data bits, parity, and stop bits for a modem connection. For instance, if you are dialing in to a host system that expects a connection with 7 data bits, even parity, set it here (setting it on the serial port will have no effect).

5.4.3   Error Control

Under the Advanced button, you can enable or disable error control. We recommend this be enabled. If enabled, you can also enable or disable compression. Compression will give you higher throughput, but may be reliable only if flow control is set correctly.

5.4.4   Flow Control

Also under the Advanced button, you can enable flow control, and specify hardware or software flow control. We recommend this be enabled, and we have a slight preference for hardware flow control. Note that this setting governs the flow control between your PC and its modem. There can also be flow control problems at the host end. See below.

5.5   Troubleshooting Modem Connections

There are, of course, many things that can go wrong with a modem connection. We'll try to offer some tips here on the most common ones. Several of these make use of Anzio's Diagnose/Debug Communication menu item.

5.5.1   Failure to Establish Connection

If Anzio fails to connect your modem to the host-end modem, it will give you a message box indicating, for instance, that it received a busy signal. If that doesn't give you enough information to troubleshoot the problem, there are a couple other things you can do.

You can click on Diagnose/Debug communication, and look at the debug information that is presented. The most useful information is likely to be at or near the bottom. This might tell you some extended error information reported back from Windows' modem support, for instance.

You may also want to listen to the phone connection. Plug a two-phone adapter somewhere between the modem and the wall jack, and plug a standard (analog) telephone into it. Then listen as the modem tries to connect. This will enable you to hear a recorded message that says you have to dial an area code, for instance.

5.5.2   Baud Rate Shift

If the modems connect, but then you see only garbage characters on Anzio's screen, you may have a problem of baud rate shift. This is caused by an incorrect configuration of the *answering* (that is, the host-end) modem. This problem can appear when a newer, faster modem is first used to connect to the host's modem.

The host-end modem has to know what baud rate to use when talking to the host machine. It SHOULD be configured to always talk to the host at a particular baud rate. The PROBLEM is when it's configured to talk to the host at whatever baud rate the modems link up at. The latter is how modems are often configured at the factory.

For instance, suppose the host system's serial line is configured for 19,200 baud. A modem is connected to that line, and people have succeeded for quite some time in connecting at 19,200. Now, you have a brand new PC, with a built-in modem that is capable of 56k. You dial in, and the two modems establish a link at 38,400. The host-end modem, being improperly configured, starts trying to talk to the host at 38,400. But the host port is still working at 19,200. So there is a baud rate mismatch, and you see garbage.

The proper solution is to configure the host-end modem to always communicate with the host at a fixed baud rate, the same rate as the host's serial line is configured for.

5.5.3   Flow Control

Flow control (also called handshaking) is a protocol by which one party to a conversation can say "hold on a minute", until it's ready to receive more data. In general, there are two kinds of flow control. Hardware flow control makes use of separate wires in the serial interface, such as the Clear-To-Send (CTS) wire. Software flow control uses certain characters (XON, XOFF) to start and stop data.

There can be issues of flow control in both directions between the PC and the host. Most often, though, it will be a problem in data coming from the host to the screen, especially when a large amount of text is sent without pausing. There is an even greater chance of flow control

problems coming to light when doing passthrough print, because the host can push out data much faster than the printer can print it. To restate, you may have a problem in flow control but it will show up only when you do passthrough print or display large amounts of data.

On a modem connection, there are really three segments: host-to-modem, modem-to-modem, and modem-to-PC. The modem-to-modem connection is handled by the error-correcting protocol of the modems, so it is generally not a problem. The other two segments are actually independent of each other.

In the host-to-modem segment, the flow control (handshaking) configured in that modem must be coordinated with the flow control used by the serial port driver of the host's operating system. For instance, if the host is running Unix, then the "stty" settings must specify the same kind of flow control as is configured into the modem.

In the modem-to-PC segment, Windows does a good job of coordinating the kind of flow control between the modem and Windows itself. The kind of flow control is chosen in the Advanced part of the Modem Properties, as mentioned above.

6   Printing

Anzio can do several kinds of printing:

- printing the current screen or window
- printing incoming text as it is displayed (*capture to print*)
- printing incoming text without displaying (*passthrough print*)
- sending an incoming file transfer directly to the printer
- printing a file from the PC's disk

The following sections describe:
- *Print Levels*
- *Choosing a Printer*
- *Setting Printer Options*
- *Choosing A Font*
- *Printing The Screen*
- *Capture To Printer*
- *Passthrough Printing*
- *Secondary Passthrough Print*
- *File Transfer To Printer*
- *Printing Files*
- *End-of-Print Flush Timer*
- *AnzioWin's Print Wizard*
- *The Print Preview Dialog*
- *Troubleshooting Printing Problems.*

6.1   Print Levels

Anzio can use several methods of writing to the printer. These methods are referred to as *print levels*.   Different print levels are required for different printing situations, especially during passthrough print.

In some situations, the host will send out just plain text, with returns and linefeeds, but no escape sequences. In that case, Anzio can use the Windows printer interface and printer driver to "draw" the text on the page, in the correct orientation, font, height, width, spacing, margins, etc.

In other situations, the host application knows (or thinks it knows) what kind of printer is connected, and it wants to be in control. It includes

escape sequences in the print data, to control the printer. In this case, Anzio must stay out of the way, passing on the data transparently.

This is an example of the need for different print levels. Following is an explanation of each Print Level that Anzio offers. These can be selected in the File/Printer Setup dialog, or with various ANZIO commands.

> ➢ AnzioWin (but not Anzio Lite) has an extra feature called Print Wizard, that will in most cases decide automatically on the print level. Print Wizard is explained below.

6.1.1   High

In high level, Anzio is in charge of layout. Various properties you have selected in File/Printer Setup will be in effect, including paper size and orientation, paper source, output bin, etc. Also, the selections in File/Printer Font will be used. Anzio will determine line spacing, lines per page, etc.

If the print level is high, and the data coming from the host computer contains printer control codes, the codes will be drawn on the page instead of being obeyed. Also, if a host program (such as Word Perfect for UNIX) is generating PostScript code during passthrough print, the PostScript code will be printed on the page, rather than obeyed. These are indications that you need to switch to a different Print Level.

6.1.2   Low

> ➢ We no longer recommend low level.

In low level, Anzio uses the printer driver, but uses a "passthrough" capability within the driver. Escape sequences are obeyed. Some printer setup settings will be obeyed. The font selection may be obeyed; which may cause problems (if a non-printer-resident font is selected).

6.1.3   Spooler

Spooler level is the recommended level when data contains escape codes.

In Spooler level, Anzio sends print data directly to the Windows spooler. Anzio and Windows are completely transparent, so escape sequences from the host are obeyed by the printer. But settings under Printer Setup and Printer Font are ignored.

In this level, if the data from the host does NOT contain escape sequences to configure the printer, then formatting (font, size, linespacing, etc.) will be up to the printer. This may mean the formatting is affected by the last job that was printed. If the printer has front panel controls, you should be able to control formatting there.

> ➢ Some low cost printers, sometimes called "Windows only printers", can not accept data at this level. With them, the only possible Print Level is High.

The Windows printer driver is not used, so it does not matter what driver is installed. The printer setup is referenced only to get the device name of the printer. However, the spooler settings in the Advanced part of the printer's control panel WILL have an effect on spooling.

6.1.4   <u>Raw</u>

Raw level is as low as it goes. In Raw level, output is sent directly to a device (or file), with appropriate handshaking and timeout management. Like Spooler level, Raw level is completely transparent, so Anzio exerts no control of formatting or other printer settings.

The Windows printer driver is not used, but is referenced to determine the port location of the printer.

> ➢ Raw printing will not work for printers attached by certain methods, such as USB, USB-to-serial adapters, network interfaces, or Novell servers.

> ➢ Raw printing will not work with "Windows-only" printers.

6.2   Choosing a Printer

To get started in printing configuration, go to File/Printer Setup. You will see this dialog box:
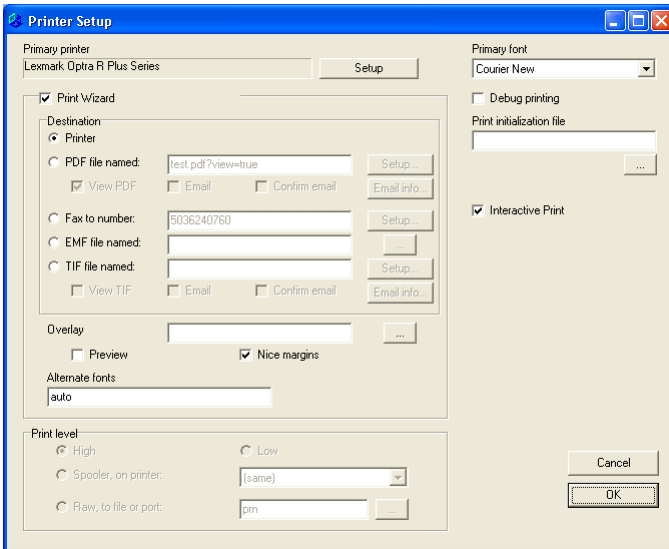
In the top left you can select a primary printer, by clicking the "Setup" button. Normally, this printer is used for all levels of printing. However, it is possible to specify a different printer to be used for Spooler-level printing, and/or a different printer to be used for Raw-level printing. These are explained below. When configured for different printers, changing the Print Level also changes what printer you will output to.

6.2.1   The Primary Printer

To select the primary printer output device, use the Setup button shown above, or the CHOOSEPRINTER command. The printer can be connected to another PC on a network, connected directly to a network, or connected directly to this PC via a parallel port, a serial port, or a USB port. Anzio simply sends data to the selected printer by name, and Windows does the rest.

The primary printer can be any printer that has a Windows driver, including "Windows-only" printers (if Print Level is High).

> ➢ We generally recommend that you do NOT use the printer driver "Generic".

6.2.2   The Spooler Printer

To specify a different printer to be used when Print Level is Spooler, use File/Printer Setup, and select it opposite the "Spooler" selection; or use

the SPOOLER command. The printer is identified by its Windows name. Select "(same)" to use the primary printer.

➢ UNcheck the "Print Wizard" box to get access to Print Level settings.

6.2.3   The Raw Printer

To specify a different printer to be used when Print Level is Raw, use File/Printer Setup, and enter a name in the box opposite the "Raw: selection; or use the RAW-SETUP command. The printer is identified by the name of its port, such as LPT1 or COM2.

➢ UNcheck the "Print Wizard" box to get access to Print Level settings.

To use the primary printer for raw printing, set this box to "(same)", or leave it blank.

If a serial port is specified, you will be prompted for the baud rate and other communication parameters. These must match how the printer is configured. Settings for that port at the Windows Control Panel level will be ignored.

Once you have specified a *port name* for raw printing, you do not need to have a driver installed for that port. In fact, if a driver is installed, it may interfere with printing. In Windows NT and later, the driver will intercept the raw-level output and route it to that printer driver's spooler. This can effect the timing of your printing output. You can address this by a) changing the spooler settings in the Advanced area of that printer driver; or b) deleting the printer driver.

You can also specify a disk file name, in which case Anzio will write the printout to disk. Note that if a file of the specified name already exists, it will be overwritten with no warning.

6.3   Setting Printer Options

➢ Many settings in Printer Setup apply only when Print Level is High.

In the Printer Setup dialog, click the "Setup" button next to "Primary printer". This displays the standard Windows printer selection dialog box. Use the dialog's Preferences button to change print quality, orientation, paper size, etc.

The next time you print something, Anzio will use the specified printer and settings. Unlike many Windows programs, Anzio does NOT pop up the Printer Setup dialog every time you want to print, for most kinds of

printing. However, if you have Interactive Print checked in the Printer Setup dialog, then this same dialog will be presented every time a passthrough print job is received.

Anzio saves changes to your printer settings in its settings file (normally "AnzioWin.def"). These settings will override settings made at the Windows control panel level.

6.4   Choosing A Font

➢ The font settings apply only when Print Level is High.

Use File/Printer Font to choose a font and size for printing. Or, you can choose the font name in the "Primary font" pulldown in Printer Setup.

You can use any font on your PC, but some will work better than others. Fixed-width fonts, such as Courier New, will line up properly in columns. TrueType fonts will give you more options for size. Printer-resident fonts will generally print faster than TrueType fonts.

We recommend "Courier New" for most applications. However, if you are printing to a dot matrix printer, this may make your printing painfully slow. In this case, choose a font that is resident in your printer; these will generally have a printer icon beside them in the font selection dialog. For instance, in an Okidata 320, you might choose "Courier 10CPI". You might need to choose different names in order to get different sizes.

The font you are choosing here is the "primary font". If *Print Wizard* is on, and Anzio encounters international or other special characters that are not in the font selected, it will attempt to find an appropriate font, for just those characters.

The font size you choose will directly affect horizontal spacing (therefore characters per line) and vertical spacing (therefore lines per page). An 80-column page should fit with 12 point text, or 10 characters per inch (*cpi*), on a page with at least 8 inches of printable space. For a 132-column page, you will probably want to change your font size down to 7 point.

➢ *AnzioWin's Print Wizard*, page 73, if enabled, will scan incoming printer text, then fit the text to the page automatically. This means that your chosen font size will not have an effect.

6.5   Printing The Screen

To print the currently active screen, use File/Print Screen, ALT P , PRTSC ,
or the PRINT command, or select "Print Screen" from the popup mouse
menu. If you have highlighted a portion of the screen using the mouse,
only the highlighted text will be printed.

The PRINT command can also print any rectangular region on the
screen. See the reference section.

> ➢ Printing the screen does not automatically eject the page – see *End-of-Print Flush Timer*, page 72.

> ➢ To print any buffered printer data and eject the page, use File/Eject, or the EJECT command.

6.6   Capture To Printer

Anzio can send incoming screen text to the printer as it arrives ("*capture
to print*"). You can start and stop print capture at any time, without
affecting the host.

To start capture to print, checkmark the menu item Transfer/Capture to
Printer, or execute the command CAPTURE WPRN. To stop, uncheck
menu item Transfer/Capture to Printer, or use CAPTURE OFF.

6.7   Passthrough Printing

Anzio supports *passthrough print*, also called *transparent print*, *local
print*, or *slave print*, to send incoming host data directly to the printer,
instead of to the screen.

Passthrough print is controlled by the host system, using terminal-type
specific escape codes. Anzio recognizes passthrough print codes for all
the emulated terminal types. When Anzio receives a passthrough print
job from the host, it will check the setting of the Interactive Print setting in
File/Printer Setup. If this setting is ON, the Printer Setup dialog will pop
up, and you can choose the various options for print processing, such as
which printer to use, whether to create a PDF, etc. If this setting is OFF,
print handling will be done in the way that Printer Setup was last
configured.

If your host does not provide passthrough print, and you would like to
use it, contact Rasmussen Software for more information.

6.8 Secondary Passthrough Print

When Anzio is emulating an ANSI-style terminal (VTxxx, SCOANSI, LINUX, AT386, or ANZIO), it supports a secondary passthrough print channel. Secondary passthrough print always goes to the Raw Printer as described above. This provides a way to support multiple printers, especially for point-of-sale environments.

To initiate secondary passthrough print from the host, send out an escape character, followed by "[7i". To turn OFF secondary passthrough print, send out escape followed by "[6i".

6.9 File Transfer To Printer

To send data from the host to the printer, using Kermit or similar transfer protocols, specify the PC output file as "WPRN". When Anzio sees this special name, it will send the received data to its print handler.

Or, in the FTP/SFTP dialog, you can right-click a file and choose "Print".

6.10 Printing Files

With Anzio's command interface, it is possible to print files that exist on a local disk, on a networked drive, or on an FTP, HTTP, or HTTPS server. Just use the PRINTFILE command, or use the COPY command to copy a file to "WPRN".

6.11 End-of-Print Flush Timer

When you print from Anzio, the output does not necessarily include page feeds (paper eject).

Printer data is saved up in a *print job*, until the job is *flushed* (sent) to the printer or print spooler.

To finish printing a page or job, use File/Eject. See also the FLUSH and EJECT commands.

You can set a delay time to use between the last printed character and an automatic eject. Use File/Flush Timer, or the FLUSHTIMER command. The default FLUSH TIMER is 5 seconds. If something has been sent to the printer, and 5 seconds go by without anything more being sent to the printer, Anzio assumes you are done printing, and it flushes the job. Five seconds after a screen print, or 5 seconds after your last passthrough print data comes in, the data is printed on the printer.

Note that the passthrough print protocol does not have a mechanism by which the host can tell Anzio that it is finished with a print job. That is why the FLUSH TIMER setting is critical. If passthrough print data arrives sporadically from the host, with pauses, you may need to increase the FLUSH TIMER, to prevent Anzio from breaking the one print job into many. Conversely, if you need the print job to be processed immediately, set FLUSH TIMER lower, as low as 1 second.

You can disable this feature by setting FLUSH TIMER to 0. In this case, the print job will be closed by File/Eject, an EJECT or FLUSH command, or when you exit Anzio.

Finally, you can set the FLUSH TIMER to any negative number. That tells Anzio to flush the job immediately.

6.12   AnzioWin's Print Wizard

AnzioWin's Print Wizard feature is intended to solve several problems, as well as provide greater flexibility in print output. You can turn Print Wizard on by checking the Print Wizard box in File/Printer Setup.

Incoming print data can be structured many different ways. The host may assume certain printer settings, such as paper size, or it may try to set them with "standard" escape codes. Print Wizard will attempt to deal with a wide variety of these situations intelligently.

Print Wizard also lets you control and modify the output. You can apply a form overlay to a print job, for instance. Or you can generate a PDF or fax from your print job, with no additional software.

6.12.1   How Print Wizard Deals with Text

When the Print Wizard is on, it will store and analyze data to be printed, before beginning to print. If the file contains plain text, Print Wizard will determine the maximum number of columns per line, while dealing with tab expansion, backspace-bolding, backspace-underlining, wide Far East characters, etc. It will also determine the number of lines per page, if the print data contains formfeeds, and makes some assumptions if it does not. Finally, it calculates and then uses logical values for margins and character height, width, pitch, and linespacing. The result is that data in almost any format is printed in a logical way, with no user intervention.

During its analysis, Print Wizard will notice if the file contains escape codes or PostScript source. If so, it will stop further processing and switch to spooler-mode printing, so those codes in the data can control the formatting of the printout.

However, if those escape codes are in the PCL-5 printer control language (created by HP for LaserJet printers), you have the option of having Print Wizard translate those codes. This is the TRANSLATEPCL option. This means that you can print PCL-5 print jobs on non-PCL printers; in fact, on any Windows-supported printer including "Windows-only" printers. This is a powerful added feature of Print Wizard.

> ➢ Print Wizard configuration switch for translating PCL is in the Edit/Advanced Options/Print Wizard dialog, and is explained on page 218.

6.12.2   Directing Print Wizard Output

Print Wizard also allows you to direct print output to someplace other than a printer. In the Printer Setup dialog, you can choose a "Destination". Besides "Printer", you have these options:

PDF   Print Wizard has a built-in PDF generator (no other software is required). If you choose this option, you can enter an output filename, or, if you leave this blank, Print Wizard will assign a name automatically. You also have a "View" option; if checked, Print Wizard will "start" the generated file after it is finished, which generally means it will be displayed in Acrobat Reader (or whatever program is associated with the ".PDF" file extension).

You can set other options for the PDF to be created by clicking the "…" button to the right of "PDF". Anzio will present a dialog box to let you choose various options. One option is to email the created PDF; if you check that option, the dialog box will expand to allow you to specify the contents and options of the email.

FAX   With the fax software built into Windows 2000 and later, Print Wizard can render directly to fax output. You must indicate a number to fax the document to.

If you click the "…" button opposite "Fax", you will be presented with a dialog box to let you specify various aspects of the faxint operation.

EMF   You can create an Enhanced Meta File out of the report. If the report is multiple pages, multiple files will be created.

| TIF | You can direct Print Wizard to generate a TIF file from the print data. You can specify only a filename, or click the "…" button to see a dialog to allow you to specify various other options. You can specify that the TIF should be emailed, similar to how PDFs are handled. |

The various options for Print Wizard rendering, including PCL translation and overlay printing are possible when creating these alternate kinds of output, as well.

6.12.3   <u>Other Print Wizard Options</u>

Other options for Print Wizard that can be controlled by the Printer Setup dialog include:

| Overlay | Here you can choose a file that contains a form image to be combined at print time with the data in the print job. This can be used to avoid using preprinted forms, for instance, when printing invoices, purchase orders, medical forms, etc. Many file formats are supported. Just click on the browse button to pick a file. |

| Preview | The Print Preview option allows you to see a file before it is printed. You can then adjust margins, pagination, and other printing parameters; markup the document (add comments, signatures, circles, and arrows); and control label printing. |

| Nice margins | If this is checked, Print Wizard will add some margins to the text; if not, Print Wizard will print at the highest and left-most possible point, in order to fit on preprinted forms. |

| Alternate fonts | If Print Wizard is told to print some special or international characters that are not in the Primary Font, it will look for other fonts with which to print those characters. This box controls how that is done. You can enter a comma-separated list of font names, and Print Wizard will attempt to use fonts in that order. If there is a font you want Print Wizard NOT to use, put a minus sign at the beginning of that font. End the string with "auto" to tell Print Wizard to try all other fonts on the Windows system. |

➢ There are other settings that control the operation of Print Wizard, in Edit/Advanced Options/Print Wizard.

6.12.4 <u>Print Wizard Markup Language (PWML)</u>

Print Wizard also gives you a unique ability to control the printer from a host program. Print Wizard has its own extensions to the Web's HTML markup language to specify everything from paper orientation to bitmaps. This system is called Print Wizard Markup Language, or PWML.

Print Wizard's markup language provides the programmer a way to control Anzio's printing in a device-independent manner. With markup tags, the programmer can specify, among other things, paper size, paper orientation, fonts (name, size, bold, italic, underline, etc.), form overlays, margins, etc., as well as printing bitmaps, barcodes, rectangles, and more. The PWML code can be generated in nearly every programming language, as it consists of only ASCII characters.

PWML code can also specify a full range of international (Unicode) characters. Print Wizard will use its sophisticated font substitution logic to find a font and print these characters appropriately.

6.12.5 <u>Printing HTML</u>

Print Wizard can process and print many kinds of HTML files. In fact, PWML is based on HTML, with some additional tags added.

However, there are some elements of HTML that Print Wizard will NOT handle:

> - Frames
> - Tables
> - CSS
> - Free-form text with word-wrap

> ➢ For more information on Print Wizard, see the file PrintWiz.doc that came with AnzioWin.

6.12.6 <u>The Print Preview Dialog</u>

If you have told Print Wizard to use Preview, most print jobs will generate a Print Preview dialog:
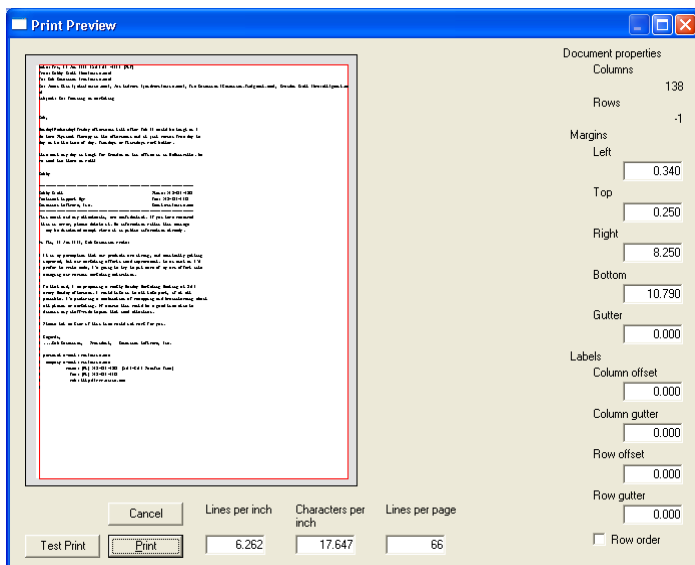
The purpose of the Print Preview is to allow you to see the initial page of the print job laid out as it will look on the paper. You can adjust the page margins and the character sizing before printing. You can also control reformatting of continuous-feed label data into full-sheet format.

If you have specified that the job should be printed with a form overlay, the form should show as part of the page image. You can't control the placement of the form image at this point, but you CAN control the placement of the characters on form.

Note that you can enlarge or zoom the Print Preview window, to see the text better. If Anzio's View/Track Window Position menu item is checked, Anzio will save the size and zoom state of this window; be sure to save your settings in order to retain this information between Anzio sessions.

> ➢ Changes made in Print Preview are not retained once the job has been printed.

6.12.6.1 Document properties

In the upper right corner, the window shows you the document properties according to Print Wizard's analysis. The Columns item shows the maximum number of columns (characters) per line. It also shows the maximum number of rows (lines) per page, IF the incoming print job has page break characters (formfeeds); otherwise, it shows "-1".

6.12.6.2 Character size

The "Lines per inch" and "Characters per inch" items determine the character size, in the vertical and horizontal dimensions respectively. You can change these, and see your change reflected in the preview page.

The "Lines per page" item indicates how many lines will be fit between the top and bottom margin. It is interrelated with the "Lines per inch" item. If the print job does not have page breaks (formfeeds), changing "Lines per page" will change the pagination of the printout.

6.12.6.3 Margins

The various margins constrain the printing. Note that the Bottom margin is measured from the TOP of the paper, and the Right margin is measured from the LEFT side, somewhat unconventionally.

The Gutter amount is additional space for binding. It is added to the left side of odd-numbered pages and to the right side of even-numbered pages.

Changing any of the margins may change the character size ("Lines per inch", "Character per inch") as Print Wizard works to fit the data within the margins.

The margins are indicated in the page display by a red outline.

> ➢ The gray area around the edges of the page image shows the unprintable area of the page, as reported by the printer driver you are using. Setting the margins to allow printing in this area could result in a loss of data.

6.12.6.4 Label dimensions

The "Labels" area controls Print Wizard's ability to convert continuous-feed label data into full-page layout. The "Column offset" breaks the page into columns, and the "Row offset" breaks it into rows. For more information, please see the separate Print Wizard documentation.

6.12.6.5 Popup menu

In the Print Preview window, a right mouse-click brings up the popup menu:

```
   Copy
   Start labels here

✓  Allow ink markup
   Clear ink
   Pen color                    ▶
   Use as eraser
   Eraser style                 ▶

   Print ink on first page only
```
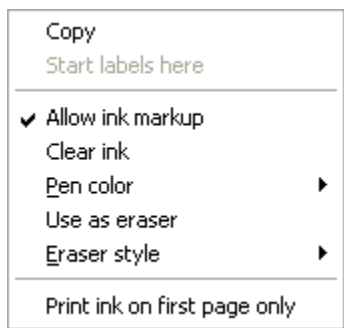
The "Copy" item will copy the page image to the Windows clipboard, including any markup.

"Start labels here" is available if the label feature is being used. It allows you to start the printing at a particular label.

The "Allow ink markup" section controls markup, which is possible with most Windows XP systems and all TabletPC and Vista systems. When enabled, you can draw on the page image using the mouse or the stylus, change pen colors, and erase your marks. This allows you to notate documents, sign purchase orders, etc. Any markup is included in an image copied to the clipboard.

When you click the Print button, and Print Wizard proceeds to print the document, any markup will be printed on every page, or, if the last menu item is checked, on the first page only.

6.12.7 The Test Print Button

The button labeled "Test Print" will print only the first page of the print job, and you will stay in the Print Preview window. If necessary, you can adjust your margins or other parameters, and then Test Print again. When everything is satisfactory, you can proceed with "Print".

6.13   Print Initialization Files

In some situations, the user needs to specify certain printing parameters or customize the printing of passthrough print jobs. For these situations, Anzio allows a *print initialization* file.

The name for the print initialization file is given in File/Printer Setup, as "Print initialization file". When Anzio receives a passthrough print job, it looks to see if this file exists. If it does, the contents of this file are **prepended to**, that is, inserted onto the beginning of, the print data, before it is routed to Anzio's print handler (or Print Wizard).

This file is a plain text file, and can be constructed in any text editor, such as "Notepad". Any text, control codes, or markup tags will be sent to the print handler ahead of the passthrough print job. Text, for instance, would be printed at the beginning of the print job. Control sequences (escape codes) will be sent to the printer (assuming the print level is not High).

The most common use of the print initialization files is to include PWML tags to change the formatting of the printout. These tags might set orientation, paper size, multi-column labels, font, lines per page, etc. PWML tags can also specify a bitmap logo to be printed at the beginning of the job.

See the file PRINTWIZ.DOC for more information.

6.14   Troubleshooting Printing Problems.

There are many ways that printing can go wrong. This section will help you fix some of the more common ones.

First, go to File/Printer Setup and turn on "Debug printing". This will cause Anzio to pop up an information window whenever it processes a print job. That window gives the most information when Print Wizard is on, but can give some useful information in any case. If you don't understand its output, you can email it to Rasmussen Software as part of a support call.

6.14.1   When it Doesn't Print At All

1. Make sure you can do a test print from the Windows printer driver. If this doesn't work, the problem is outside of Anzio.

2. Try printing the screen from Anzio, with Print Level set to High.

3. During passthrough print, if Debug Printing is on, and the debug box doesn't pop up, then Anzio is not actually getting the print job. Passthrough print is not happening. There is a problem in the configuration of the host system.

4. If data goes to the screen when it should be going to passthrough print, then the host system may not have the correct code to turn printing on and off in its terminfo, termcap, or equivalent file.

5. If you're trying to print in Raw mode, try Spooler instead.

6. If the Debug Printing information indicates that "data contains escape codes", the printer you are using may not understand

those print codes. For instance, the host may be sending PCL-5 codes, but you aren't using a PCL-5 printer. Try turning on "Translate PCL", in Edit/Advanced Options/Print Wizard.

6.14.2   When it Prints, but Wrong

1. If you're getting extra characters, especially at the beginning of the job, those may be escape sequences sent by the host. Try setting Print Level to Spooler, or use Translate PCL.

2. Use the correct printer driver for your kind of printer. If it's a new printer, check to see if there's a later printer driver available from the manufacturer's web site. If there's still a problem, try using a driver for an older related printer, such as a LaserJet 5 driver for any HP PCL-capable laser printer.

3. Do not use a Generic driver if at all possible.

4. If Print Level is not High, and print layout is wrong, the problem is in the host application. In these levels, Anzio is completely transparent. However, Print Wizard may be able to help – turn it on.

5. If passthrough print data is intermixing with screen data, then the passthrough print mechanism on the host is poorly designed. See our web site for help.

6. If the print data starts OK, but subsequent pages creep up or down the page, there is a problem with lines per page. Try experimenting with font size. If you're using Print Wizard, either adjust the host system to send 66 lines per page, or use a print initialization file (see above) to tell Print Wizard how many lines per page the host data contains.

7. If your host system lets you tell it what kind of printer you have, try telling it you have a "generic" or "no codes" printer, so the application will send out plain text (without control sequences). Then turn on Print Wizard. Print Wizard will automatically fit each report to the page.

8. If international characters or diacritics are wrong, Anzio may not be set to the correct character set. Go to Communicate/Character Sets/For Printer/8-bit, and set it to match the encoding scheme used in the data.

9. If printing is too slow on a dot matrix printer, and Print Level is High, try choosing a font that is resident in the printer (one without a "TT" icon).

10. If there is a long delay before printing starts, try going to File/Flush Timer, and setting it to 1.

➢ Solutions that involve Print Wizard will not be available if you are running Anzio Lite; Print Wizard is a feature only in AnzioWin. You may want to try downloading and running a demo version of AnzioWin, with Print Wizard turned on. If it solves your problem, you might consider upgrading to AnzioWin.

Part II   Data Capture and File Transfer

7   Data Capture

Anzio lets you capture screen data to the printer or to a file. This captured data can be *static*, that is, what's on the screen at any point in time, or *dynamic*, where it's captured as it is received from the host.

The next section, *File Transfer*, describes how to transfer files to and from the host.

The following sections describe:

- *Static Capture*
- *Dynamic Capture*
- *Control Characters During Capture*
- *Character Set of Output File*

7.1   Static Capture

This section describes capturing data that is already on the screen. You can capture screen data as characters (KEEP), as data columns (PICK), or as printed text (PRINT). The following descriptions refer to Anzio's commands:

KEEP   Reads the screen, or an area of the screen, and saves the characters to the current output file. Use OPENO first to open an output file. Enter the KEEP command to save the entire screen, or use KEEP F7 to select a part of the screen.

PICK   Reads a highlighted area of the screen as data columns, and saves them to the current output file. PICK writes each row of data in the standard comma-separated-values (CSV) format, for use with other programs.

PRINT   Prints all or part of the live screen, except the status line You can also use PRINT F7 to select a part of the screen.

Through the menu system, you can also do the equivalent of PRINT: use File/Print Screen or ALT P to print the screen, or press PRTSC .

Through the menu system, you can also do the equivalent of KEEP. First, open a file using File/Open New File to Receive. Then, do Edit/Copy Screen to Output File.

7.2 Dynamic Capture

This section refers to capture of characters as they are coming from the host system.

With Transfer/Capture to Printer, you can tell Anzio to capture incoming data to the printer. Data will go to the Print Wizard, or to the printer at a particular Print Level, as configured. Note that the print job will not finish printing (and may not even start printing) until you turn OFF Capture to Printer (Flush Timer is ignored here).

With Transfer/Capture, you can capture to a file. Anzio will allow you to choose the location and name of the file to be created, then will start capturing to it.

7.3 Control Characters During Capture

When you are capturing data coming from the host, often the readable data will also contain control sequences, such as for cursor positioning, that you don't want to capture. However, there may be control characters that you need to capture also. So you have the option of specifying which, if any, control characters to capture, in Edit/Advanced Options/General. There you can set "Capture control characters" to "None", "All", or "All except escape sequences". We suggest experimentation.

> ➢ Certain settings will alter how the data is formatted on the screen.

7.4 Character Set of Output File

If data being captured is other than plain ASCII, there is the question of what character set (encoding scheme) should be used in writing it to the file. Go to Edit/Advanced Options/General, and see "Character set for output file". Your options are "ISO", "OEM", and "UTF-8".

For the best capture of international characters, choose "UTF-8". The resulting file can be imported in Microsoft Word, for instance, if you identify it as "encoded text" with encoding "UTF-8".

8   File Transfer

Anzio provides several methods to transfer files between your PC and the host. The previous section described how to capture screens of data to a file.

This section shows how to transfer files in either direction:

- PC to host, called "upstream" or "upload"
- Host to PC, called "downstream" or "download"

In many cases we can transmit data directly into existing software on the host system, and it will think we are typing it. In nearly all cases we are limited to passing ASCII data only.

A *simple upload* involves sending sequential ASCII records upstream. Most text files can be sent to the host with this method. See:

- *UNIX Simple Upload*, page 87

For some kinds of file transfer you will need special host software. The Anzio distribution diskette includes several simple file transfer utility programs for different hosts. See:

- *UNIX File Transfer*, page 86
- Anzio can also send and receive files using the standard protocols *Kermit* and *Zmodem*. Each of these protocols requires that the host have the corresponding software installed.

The following sections describe:

- *UNIX File Transfer*
- *Using Kermit*
- *Using Zmodem*
- *C332 File Transfer*
- *FTP*
- *SFTP*

8.1   UNIX File Transfer

This section describes several file transfer methods for UNIX operating systems:

- *UNIX Simple Upload*
- UNIX Simple Download Script (download)

- UNIX Download Utility (Send-PC.C)
- Using Kermit

> The utility programs Send-PC and Recv-PC are provided in C source form. To use a utility, you must first upload the source file to the host. Then, you will need to compile the source code into an executable program. See the example under *UNIX Download Utility (Send-PC.C)*, page 88.

8.1.1   <u>UNIX Simple Upload</u>

You can upload PC text files directly to a UNIX or Linux host.

First, tell UNIX/Linux to copy incoming characters to a file:

```
cp /dev/tty filename⏎
```

Then, tell Anzio to open and transmit the PC file, following it with UNIX end-of-file (CTRL D):

```
CTRL SHIFT F OPENI pcfilename ⏎
CTRL SHIFT F TRANSMIT TRAILER CTRL D ⏎
```

8.1.2   <u>UNIX Simple Download Script (download)</u>

One of the utility files provided is a UNIX shell script called download. This script can download one or more text files at a time to the PC. Unlike the Send-PC utility described below, the download script does not need to be compiled.

1. Use the *UNIX Simple Upload* procedure to upload the download file to the UNIX host.

2. Tell UNIX to make the file executable:

```
chmod +x download⏎
```

3. To download a file, tell UNIX:

```
download filename⏎
```

download sends the indicated file(s) to Anzio, using each host file name for the PC file name. *filename* can contain UNIX wildcard characters, for example, to download all C source files in the current directory, tell UNIX:

```
download *.c⏎
```

> ➢ `download` will overwrite an existing PC file of the same name without warning.

8.1.3   UNIX Download Utility (Send-PC.C)

The SEND-PC program will transfer virtually any UNIX text file down to the PC. You will need to upload the C source file Send-PC.C and compile the program on the host.

First, tell UNIX:

```
cd  ⏎
cp /dev/tty send-pc.c⏎
```

Then, tell Anzio:

```
[CTRL] [SHIFT] [F] OPENI SEND-PC.C  ⏎
[CTRL] [SHIFT] [F] TRANSMIT TRAILER [CTRL] [D] ⏎
```

Wait until the transfer is finished, then tell UNIX to compile send-pc.c into the object file "send-pc":

```
cc send-pc.c -o send-pc⏎
```

If you do not have a C compiler ("cc"), we may have provided you with objects (executable programs) for your particular UNIX system. Check the README.TXT file on the distribution diskette.

To use Send-PC, tell the host:

```
send-pc unixfilename pcfilename  ⏎
```

The program will delete any existing *pcfilename*, create a new one, and send the *unixfilename* down.

8.1.4   UNIX Upload Utility (Recv-PC.C)

The program RECV-PC.C is included for use with PC text files. You will need to upload and compile Recv-PC.C as described above.

To transfer a file, tell UNIX:

```
recv-pc unixname pcname⏎
```

8.2   Using Kermit

The Kermit file transfer protocol is available for most UNIX, Linux, and similar systems, on the Internet. It may be installed already on your system. Kermit allows transfer of many kinds of files between many

kinds of systems, with error checking, data compression, and more. Contact us for information on obtaining Kermit for your UNIX machine.

> ➢ Anzio can respond to an incoming Kermit file download request automatically, if so configured. Go to Transfer/Transfer options, Select protocol to configure, choose Kermit, click Configure, and checkmark Kermit auto reception.

Anzio provides several Kermit-specific commands: KSEND, KRECEIVE, and KCOMMAND. These commands are described beginning on page 111.

Kermit on UNIX should come with instructions, but because it is not especially user-friendly, so we'll give some pointers here:

PARITY
If your UNIX system is set up for even parity, you will need to set up Kermit that way too. To do so on startup, tell UNIX:

`kermit -p e`⏎

Or, once Kermit is running, you can give it the command:

`set parity even`⏎

Finally, it is possible to put the "set parity even" command in a file named ".kermrc" in you user home directory.

TEXT vs. BINARY
UNIX Kermit must know whether files being transferred are TEXT (which translates between **linefeed** for end-of-line on UNIX and **return-linefeed** for end-of-line on Windows) or BINARY (which does no translation). To see its current setting, tell UNIX Kermit:

`sh`⏎

To set it , tell UNIX Kermit:

`set file type x`⏎

where *x* is `text` or `binary`.

SERVER MODE
For less confusion, we recommend putting UNIX Kermit in "server" mode, by telling it:

```
server⏎
```

Then, give it commands using Anzio's Kermit commands.

To terminate the server mode, tell Anzio:

`CTRL` `SHIFT` `F` KCOMMAND FINISH⏎

To terminate Kermit itself, tell Kermit:

```
quit⏎
```

8.3   Using Zmodem

Zmodem, like Kermit, is available for most host systems, and may be on your system already. It is usually in the form of a sending program ("sz") and a receiving program ("rz").

> ➢ Anzio can respond to an incoming Zmodem file download request automatically, if so configured. Go to Transfer/Transfer options, Select protocol to configure, choose Zmodem, click Configure, and checkmark Zmodem auto reception.

To download a file using Zmodem, give the host the command

```
sz filename⏎
```

To upload a file using Zmodem, first make sure the host system is at a shell prompt. Then go to Transfer/Zmodem/Send, and choose a file to transfer.

For more tips on Zmodem file transfer, see our web site.

8.4   C332 File Transfer

Anzio, when in C332 emulation, also supports a special file transfer protocol used by some Versyss systems. These transfers are controlled by the host system.

8.5   FTP

AnzioWin includes two different FTP client capabilities, one interactive and one command-driven.

> ➢ An FTP transfer requires you to have a network connection to the host system. It will not work over a serial or dial-up modem connection.

8.5.1   Interactive FTP

The Transfer/FTP menu item, or the "FTP" Anzio command, will initiate an FTP connection to the same server that Anzio is currently connected to, with your current username and password (if Anzio knows them).

The Transfer/FTP (different host) menu item, or the "FTP NEW" Anzio command, will initiate an FTP connection to an arbitrary host system. It presents first a dialog box that asks for the host name (or IP), the port, the username, and the password. Then it connects.

In either case, you will then see a dialog box that allows you to interactively move files between the host system and the PC.

8.5.2   Programmed FTP

Anzio provides commands to programmatically perform FTP operations. These commands can be typed into Anzio's Function prompt, or they can be programmed into a macro or sent from the host as an Anzio command. That means an FTP transfer can be done under program control (or macro control), without user intervention.

The commands for this are FTPGET, FTPPUT, and FTPDELETE, and also their "/S" variations. See the reference section.

8.6   SFTP

AnzioWin includes two different SFTP client capabilities, one interactive and one command-driven.

> ➢ An SFTP transfer requires you to have an SSH network connection to the host system. It will not work over a serial or dial-up modem connection, or another type of network connection.

8.6.1   Interactive SFTP

The Transfer/SFTP menu item, or the "SFTP" Anzio command, will initiate an SFTP connection to the same server that Anzio is currently connected to, as the same user.

You will then see a dialog box that allows you to interactively move files between the host system and the PC.

8.6.2 Programmed SFTP

Anzio provides commands to programmatically perform SFTP operations. These commands can be typed into Anzio's Function prompt, or they can be programmed into a macro or sent from the host as an Anzio command. That means an SFTP transfer can be done under program control (or macro control), without user intervention.

The commands for this are SFTPGET, SFTPPUT, and SFTPDELETE, and also their "/S" variations. See the reference section.

Part III   Command and Control

9   Command Reference

This section lists all Anzio text commands, first summarized by type, then described in alphabetical order. Anzio commands can come from three sources:

- Anzio's command line, where you typed the command – see *Entering Commands*, page 34.

- A defined key macro – see *Defined Keys and Macros*, page 45.

- From the host computer – see *Sending Anzio Commands From The Host*, page 202.

The following sections describe:

- *Commands Not in Anzio Lite*
- *Variable Substitution*
- *Alphabetical List of Commands*

9.1   Commands Not in Anzio Lite

AnzioWin is the fully-featured product. Anzio Lite is a less expensive product that does not include some features.

Following is a list of commands that are NOT in Anzio Lite:

> BOX
> CAPTURE CODED (do allow some forms of capture)
> CAPTURE LONG
> CAPTURE ON
> CLOSEI
> CLOSEO
> DEFINE
> DELAY/S
> DELETE
> DIR/S
> DROPOUT
> EMAILSETUP
> FAXSETUP
> FILL
> FTP
> GETCOOKIE
> IMAGEBOX
> KEEP

LAUNCH
MCISENDSTRING
MENUBAR
OLE (all forms)
OPENE
OPENI
OPENO
PDFSETUP
PICK
PLAYMEDIA
PREVIEW
PRINT (some forms)
PRINTER
PRINTFILE
REPLACE
READLINE
RECEIVE (allows receive quiet)
RECORD
RETRANSMIT
RUN
SEND (part of screen)
SETCOLOR
SETCOOKIE
SLEEP
SPEAK
SSHAUX variations
SSHPRINT variations
SSHPORT variations
TAB
TIFSETUP
TIMEOUT
TOOLBUTTON
TRANSMIT
TYPE
WINDOW *<rect>*
WINDOWCLOSE
WRITE
XN

9.2   Commands By Type

This section lists Anzio's commands by type:

▪ *Operator Preference Items*

- *Communication Parameters*

- *File Transfer*

- *Local Processing*

- *Printing*

- *Macro Processing & Key Mapping*

- *Interaction with Host and User*

- *Miscellaneous*

Commands are described in alphabetical order beginning on page 101.

9.2.1   <u>Operator Preference Items</u>

The following commands are items that a user might do to configure Anzio to meet individual preferences.

ALLOW-QUIT	Specifies whether Anzio can be terminated locally
BEEP	Turns beep on or off
BEEP IDLE	Beeps when idle
BMP	Sets a background bitmap on the screen
COLOR	Sets operating colors/attributes (old method)
COLOREDIT	Sets operating colors/attributes (current method)
CURSOR BLINK	Allows non-blinking cursor
FONT	Sets the screen font and size
FONT-ALT	Sets alternate fonts to be used for international characters
GAUGE	Shows gauge line at bottom of screen
JUMP	Sets jump scrolling
MENULANGUAGE	Sets the language for Anzio's menus, prompts, etc.
PALETTE	Sets the group of available colors and attributes
PITCH	Sets beep pitch, sound, speakers
REGSSH	Registers Anzio as the PC's standard SSH client
REGTELNET	Registers or unregisters Anzio as the PC's standard telnet client
SCREENMODE	Sets various screen modes

SCREENMODE/S	Reports screen info to host
SETKEYBOARD	Sets the keyboard input language
STATUS LINE	Turns status line on or off
TITLE	Sets window title
TOOLBAR	Configures the on-screen toolbar
TOOLBUTTON	Configures a button on the toolbar
TRACK-WINDOW	Should Anzio save its window position?

9.2.2   Communication Parameters

The following commands configure Anzio for its communication with the host system.

7E1, 8N1, etc.	Shortcuts to set standard data bits, parity, stop bits
AUTO-LF	Sets auto-linefeed
AUTO-CONNECT	Specifies whether to connect on startup
AUX	Turns on/off aux input
AUX-BIDI	Puts the AUX port into bidirectional mode
AUX-DEBUG	Turns on/off debugging of aux communication
AUX-READ	Read aux input until a certain string
AUX-SETUP	Configures aux input
AUX-WRITE	Write output to aux port
BACKSPACE	Configures the Backspace key
BAUD	Sets the baud rate
CHARSET[/S]	Sets character set translation
COMMTYPE	Sets communication type, serial or network
DATA BITS	Sets the number of data bits
DELAY	Sets the delay between characters
DELAY/S	Reports DELAY settings
FULL DUP	Sets full-duplex mode
HALF DUP	Set half-duplex mode
HOLD	Suspends host output
IGNULL	Ignores null characters

INTERPRET	Calls communication diagnostics
LINE DELAY	Sets the delay for line turnaround
LOCK	Sets the keyboard locking protocol
LOGIN-WIZARD	Specifies whether Anzio will attempt to log in automatically to the host system
MONITOR	Communication diagnostics
PARITY	Sets the parity type
PORT	Sets the communication port number
RECONNECT	Reconnect on line drop?
RESET	Resets the Anzio terminal
SCROLL	Saves erased data
SCROLL-LOCK	Does the ⌨SCROLL LOCK key suspend host output?
SSHAUX-ADD	Add a backchannel AUX serial port
SSHPORT-ADD	Add a local port forwarding tunnel
STOP BITS	Sets the number of stop bits
TAB	Defines tab stops and the character sent for Tab
TERM	Sets the terminal emulation type, such as VT100
TERMNAME	Sets the terminal type that is reported to host
UPPERCASE	Changes keystrokes to upper case

9.2.3   File Transfer

The following commands perform file transfer between the PC running Anzio and a host system, or configure that transfer.

ASEND	ASCII mode upload
CAPTURE	Captures incoming screen data to a PC file
C332_SEND	Upload a file via the C332 protocol
CLOSEI	Closes input file, optionally reporting to host
CLOSEO	Closes output file, optionally reporting to host
DELETE/S	Deletes a file and reports to host
DOWNLOAD-DIR	Sets the preferred directory for downloads
DOWNLOAD-LOCK	Restricts downloads to a certain directory

FIND/S	Finds a PC file name and sends it to host
FINDNEXT/S	Finds next PC file name and sends it to host
FTP	Does an interactive FTP session
FTPDELETE[/S]	Delete a file via FTP
FTPGET[/S]	Download a file via FTP
FTPPUT[/S]	Upload a file via FTP
KCOMMAND	Sends a Kermit command
KEEP	Copies part of screen to a file
KRECEIVE	Receives a file with Kermit
KSEND	Sends a file with Kermit
OPENE	Opens a PC file in extend mode for output from host
OPENI	Opens a PC file for input to host, optionally reporting to host
OPENO	Opens a PC file for output from host, optionally reporting to host
PICK	Picks data from screen columns for spreadsheet
PURGE	Clears the file transfer buffer
RECEIVE CODED	Receives a file coded for PC-to-PC transfer
RECEIVE QUIET	Doesn't display file transfer
RETRANSMIT	Retransmits last element
RRECV	Receives a file from the host using the configured default protocol
RSEND	Sends a file to the host using the configured default protocol
SENDFILE	Sends a file using the default protocol
SFTP	Does an interactive SFTP file transfer session
SFTPDELETE	Deletes a file on the host via SFTP
SFTPGET	Receives a file from the host via SFTP
SFTPREGET	Receives a file if necessary from the host via SFTP
SFTPPUT	Sends a file to the host using SFTP
TIMEOUT	Sets the file transfer timeout

TRANSFER-OPT	Brings up the Transfer Options dialog box
TRANSMIT	Provides simple file upload to host
WRITELN	Write text with CR/LF to output file
XN	Sends next TRANSMIT record
XRECEIVE	Receives a file using Xmodem
XSEND	Sends a file using Xmodem
YRECEIVE	Receives a file using Ymodem
YSEND	Sends a file using Ymodem
ZRECEIVE	Receives a file using Zmodem
ZSEND	Sends a file using Zmodem

9.2.4   Local Processing

The following commands can manipulate the local PC.

CALC	Invokes the built-in calculator
CD	Changes the logged disk/directory
CLONE	Create a duplicate Anzio session
COPY	Copies one PC file to another, optionally sending result to host
DDE (various)	Interact with other programs
DELETE	Deletes a file from PC, optionally ignoring "file not found"
DIALCALL	Dial a voice call
CALLDIGITS	Dial additional digits
DROPCALL	Terminate a voice call
DIR	Lists a PC disk directory, optionally sending to host
DIROUT/S	Prompt the user for an output file
DIR-TERMINATOR	Configure DIR commands
EJECTMEDIA	Closes and ejects a removable drive, such as a CD or portable USB drive.
FILE-CLEANUP	Delete old files from PC
FILEINFO/S	Get information on a PC file
FLASH	Flash the task bar to get the user's attention

GETCOOKIE	Read an Internet cookie
SETCOOKIE	Set an Internet cookie
KILL	Terminates another program that Anzio started
LAUNCH	Starts another program in Windows
LOG	Changes logged disk/directory
MAPISENDFAX	Send a fax
MKDIR	Makes a directory, optionally sending result to host
OLE (various)	Interact with other programs
RENAME	Renames a PC file
REVIEW	Scrolls back data from top of screen
RUN	Runs a PC program and then returns to Anzio, optionally sending result to host
SAVE	Saves a key macro file
SCRIPT	Runs a Windows script
SWITCHTO	Switch focus to a different Windows program
TAKEFOCUS	Switch focus to this Anzio
WINSTART	Starts a Windows program
WRITE	Writes a string to the current output file

9.2.5 <u>Printing</u>

The following commands cause printing, or configuring printing and pseudo-printing operations.

CHOOSEPRINTER	Opens the Printer Setup dialog box
CHOOSEPRINTER	Report printer name to host
EJECT	Ejects a printer page
FAXSETUP	Configure print-to-fax output
FLUSH	Releases a print job to the printer
FLUSHTIMER	Sets timed release of print jobs
INTERACTIVE-PRINT	
	Sets whether the Printer Setup dialog will be presented on receipt of a passthrough print job

NICE-MARGINS	Turns on or off the setting for nice margins in Print Wizard printing
OVERLAY	Specifies a file to be used as an overlay in Print Wizard printing
PASSTHRU	Initiates passthrough print
PASSTHRU-CP[/S]	Sets character type for passthrough print
PDF-SETUP	Configure print-to-PDF output
PREVIEW	Turns on or off Print Preview
PRINT	Prints all or part of screen
PRINTDOCNAME	Specify the name of a passthrough-printed file
PRINTER	Sets the printer to use
PRINTER-SETUP	Configures PC's printer
PRINTFILE	Prints a text file
PRINTFONT	Sets printer font and size
PRINTFONT-ALT	Sets alternate fonts for printing international characters
PRINTINIT	Specifies print initialization file
PRINTLINE	Print text with CR/LF
PRINTLOW	Sets low-level print
PRINTMACRO	Specifies macro(s) for code-activated switch
PRINTTOFILE	Controls printing to a file
RAW-SETUP	Configures Windows raw printing
SPOOLER	Specifies printer to use for print level of spooler
SSHPRINT-ADD	Add a backchannel print channel
TIFSETUP	Configure print-to-TIF output
WINPRINT	Asks Windows to print a file using the associated Windows application

9.2.6   <u>Macro Processing & Key Mapping</u>

The following commands affect macros.

BASENAME	Returns filename without path
CALL	Calls a macro key as a subroutine

DEFINE	Defines a macro
EVENT	Initiates macros when certain things happen
EXPECT	Test strings for equality
INVOKE	Stops current macro and starts another macro
KEYS	Shows special and macro keys
MERGE	Merges a macro file with macros in memory
ONERROR	Macro error handling
ONSTARTUP	Run a macro when starting Anzio
READ	Reads a key macro file
RECORD	Record a macro
REPLACE	Regular expression replacements
SET	Set a variable
SHOW-COMMAND	Configure whether commands show on screen
WAIT	Waits a specified time
WAITFOR (WF)	Waits for specific incoming text until timeout
WAITQUIET	Wait until host stops sending

9.2.7   <u>Interaction with Host and User</u>

The following commands affect the user's interaction with programs running on the host system.

ADD128	Process certain special keystrokes
ANSWERBACK	Sets the Anzio terminal's answerback string
BLOCK-HIBERNATE	
	Prevents the PC from hibernating
BLOCK-SCREENSAVER	
	Prevents the screensaver from starting up
BLOCKTYPE	Sets shape of mouse highlight
BOX	Draws a box on the screen
BREAK	Sends a Break character
C332_NEXT	Sends code to host for next C332 terminal session
C332_PREV	Sends code to host for previous C332 terminal session

CLIP	Copies screen to Windows clipboard
COMPOSE/S	Create a diacritic/character combination
CONNECT	Initiate a connection with host
DATE/S	Sends the date to the host
DIAL	Dials a modem
EMAILSETUP	Configure emailing
ENV/S	Sends an environment variable to host
ENVIRONMENT	Sets a string of variables to be sent to the host upon connection
FILL	Fills a screen area with a character
HANGUP	Hangs up a modem
HIGHLIGHT	Set the Windows highlight
IMAGEBOX	Display an image file
INK	Turns on and off markup of the main screen
MAILTO	Send an email
MCISENDSTRING	Exercise Windows' multimedia functions
MENUBAR	Creates a menu
MESSAGE	Displays a message box
PASSWORD	Sets the password for the host, or sends it to the host
PASTE	Pastes Windows clipboard text
PLAYMEDIA	Play various media files
PLAYSOUND	Plays a WAV file
PROMPT-ID	Prompts the user for username and password
PROMPT/S	Prompts the user for a string
SCREENSIZE/S	Tell the host your screen dimensions
SEND	Transmit (part) of the screen to the host
SETCOLOR	Reset current color
SIGNAL	Sends a numbered signal to the host
SPEAK	Text-to-speech
TIME/S	Sends PC time to host

TYPE	Shows a PC file on screen
UNICODE/S	Create a Unicode character
USERNAME	Sets the username for the host, optionally sending the username to host
VERSION	Displays Anzio version number, optionally sending result to host
WINDOW	Opens a text window on the screen
WINDOWCLOSE	Closes a text window

9.2.8   Miscellaneous

ALLOW-1C	Control acceptance of commands from the host
DEBUGKEYS	Opens a window to diagnose keystrokes
DEFAULTS	Saves settings
END	Quits Anzio
GETPARAM	Read a parameter setting
SETPARAM	Set an arbitrary parameter
HELP	Gets help on Anzio
HOSTNAME/S	Sends host name to host
RESTART	Quit and restart Anzio
SAVEPARAM	Save a parameter setting in memory
RESTOREPARAM	Restore a saved parameter
SLEEP	Wait until certain time
STOP	Quits Anzio
WINDOW ZOOM	Maximize the window
WINDOW MINIMIZE	Make the window an icon
WINDOW RESTORE	Bring the window out of zoomed or iconic mode

9.3   Variable Substitution

Within Anzio commands, as well as some other places in Anzio, a *variable* can be replaced by its *value*. There are several kinds of variables, as will be explained.

The general syntax is:

```
${variable}
```

Whenever this is used in a command, the entire string shown above will be replaced by the *value* of the *variable*. If the variable is undefined, the string above will be deleted; that is, it will be replaced by an empty string. For example, if you enter the command

```
winstart ${anz_last_recd}
```

then Anzio will look up the value of its internal variable *anz_last_recd*, which contains the full path name of the last file received from the host, and place that into the string. So, assuming the last file received was "c:\temp\testfile.bmp", the string above would end up as

```
winstart c:\temp\testfile.bmp
```

> ➢ The curly-brace ("{") character has a special meaning inside a macro. So if you wish to include an environment variable inside a macro, it must be in form:
> $~{value}

9.3.1 Types of Variables

Anzio supports the following kinds of variables:

Windows environment variables
Any environment variable defined in the Windows environment, that is inherited by Anzio, can be used.

User variables   User commands can create an unlimited range of variables, using the SET command.

Internal variables   Anzio defines many internal variables. See below.

Special characters   Special characters, control characters, and non-ASCII characters can be encoded as variables. See below.

Key names   Any named key will translate to the sequence or macro assigned to that key.

9.3.2 Internal Variables

Variable	Value
ANZ_VERSION	Numeric version of Anzio, such as "16.0".

ANZ_COMMTYPE	The kind of connection, such as "SSH://*hostname*", "TELNET://*hostname*", "RLOGIN://*hostname*", "MODEM://*phonenum*", or "SERIAL://".
ANZ_MAC_ADDRESS	The MAC address of the network card which Anzio is using.
ANZ_MAC_ETHERNET	Similar to ANZ_MAC_ADDRESS, but if Anzio is not connected, or its connection does not involve a MAC address (such as a PPP connection), this will return the MAC address of the first Ethernet adapter it finds.
ANZ_HOSTNAME	The name Anzio has for the network host
ANZ_IP	The PC's IP address
ANZ_PASSWORD	Value set by the PASSWORD command or menu item
ANZ_USERNAME	Value set by the USERNAME command or menu item
ANZ_SETTINGS_FILE	The pathname of the settings file that Anzio is currently using.
ANZ_KEYS_FILE	The pathname of the keys file that Anzio is currently using.
ANZ_LAST_RECD	Name of the last file Anzio received during the current session
ANZ_LAST_SENT	Name of the last file Anzio sent to the host during the current session
ANZ_TITLE	Title of the Anzio window
ANZ_WINDIR	The directory that contains Windows
ANZ_SYSDIR	The "System" directory

ANZ_CURDIR	Anzio's current working directory
ANZ_DOWNDIR	Anzio's download directory
ANZ_PROGRAM	"ANZIOWIN" or "ANZIOLITE"
ANZ_PROCESS_ID	The Windows process ID of the Anzio session. This is always unique on a particular PC at a particular time
ANZ_MY_DOCS	The folder name equivalent to "My Documents"
ANZ_COMMON_DOCS	The folder name equivalent to "Shared Documents", such as "c:\Documents and Settings\All Users\Documents". Not available on all Windows installations
ANZ_APPDATA	Usually "c:\Documents and Settings\<username>\Application Data"
ANZ_COMMON_APPDATA	Usually "C:\Documents and Settings\All Users\Application Data"
ANZ_USERDATA	Usually "c:\Documents and Settings\<username>\Application Data \<program name>" where <program name> is "AnzioWin" or "Anzio Lite".
ANZ_COMMON_USERDATA	
	Usually "C:\Documents and Settings\All Users\Application Data \<program name>" where <program name> is "AnzioWin" or "Anzio Lite".
ANZ_DESKTOP	The folder name equivalent to the Windows desktop
ANZ_PROGRAM_FILES	The folder name where programs are normally stored, such as "c:\Program files"
ANZ_PROGRAM_DIR	The folder that contains the currently running Anzio program

ANZ_PROGRAM_EXE	The full pathname of the Anzio executable program
ANZ_COMPUTERNAME	The NETBIOS name of this PC
ANZ_WINNAME	The username by which the user logged into Windows
ANZ_TCPNAME	The TCP/IP name of this PC
ANZ_LAST_LAUNCH	The status code of the last program launched by Anzio: "running", "error", or the numeric code given when the program quit
ANZ_HIGHLIGHT	The text that the user has highlighted with the mouse, on the Anzio screen
ANZ_CURSOR_X	The X location of Anzio's cursor (starting at 1)
ANZ_CURSOR_Y	The Y location of Anzio's cursor (starting at 1)
ANZ_DROP_NAME	The name of the file dropped onto Anzio's Window
ANZ_DROP_X	The X location of the point where a filename was dropped onto Anzio's window
ANZ_DROP_Y	The Y location of the point where a filename was dropped onto Anzio's Window
ANZ_ENVIRONMENT	The contents of Communicate/Environment, with embedded variables translated
ANZ_WINDOWS_VERSION	The version of Microsoft Windows, as *major.minor*
ANZ_WINDOWS_VERSION_EX	The complete version of Microsoft Windows, as *major.minor.build.servicepack*
ANZ_MOUSE_X	The column number (starting at 1) at which the mouse was clicked.

ANZ_MOUSE_Y	The row number (starting at 1) at which the mouse was clicked.
ANZ_MOUSE_WORD	The word on the screen where the mouse was clicked.
ANZ_CALLERID	Contains the phone number of an incoming voice call, if detected.
ANZ_CALLERID_NAME	Contains the caller's name of an incoming voice call, if detected.
ANZ_IS_REMOTE	Returns "1" if Anzio is running in a Terminal Services or remote desktop environment, "0" if not.
ANZ_REMOTE_IP	The remote client's IP address.
ANZ_REMOTE_NAME	The remote client's machine name.
ANZ_REMOTE_STATION	The station/session name.
ANZ_REMOTE_SESSION	The session number.

9.3.3 <u>Special Characters</u>

Variable	Value
$	A literal "$". Thus "${$}" will return an actual dollar sign
^*c*	Where *c* is a character, this returns the control-equivalent of that character. So for instance "${^a}" will return a control-a.
nn	Where *nn* is numeric, this will return a character with the decimal value of *nn*. Values above 127 are treated as Unicode.
H*nn* or X*nn*	Where *nn* is a hex number, this will return the character with the hex value of *nn*. Values above 7F are treated as Unicode.
O*nn*	Where *nn* is an octal number, this will return the character with the octal value of *nn*. Values

above 127 are treated as
Unicode.

9.4   Alphabetical List of Commands

This section describes each command in Anzio, in alphabetical order.
Each entry begins with the command name and syntax, followed by a
description.

9.4.1   <u>Syntax Conventions</u>

▪ A vertical bar | indicates alternate commands, for example:

```
BACKSPACE 8 | 127
```

▪ means that you can enter BACKSPACE 8 or BACKSPACE 127.

▪ Square brackets [ ] indicate optional parameters:

```
AUTO-LF [ON]
```

▪ where AUTO-LF is equivalent to AUTO-LF ON.

▪ *Italics* indicate a user-specified string or number:

```
ANSWERBACK string
```

▪ Where a string must contain an embedded space, such as in a
filename, put double-quotes around the entire string.

9.4.2   <u>Commands</u>

7E1

Shortcut for DATA BITS 7, PARITY EVEN, and STOP BITS 1.

7E2

Shortcut for DATA BITS 7, PARITY EVEN, and STOP BITS 2.

7M1

Shortcut for DATA BITS 7, PARITY MARK, and STOP BITS 1.

7M2

Shortcut for DATA BITS 7, PARITY MARK, and STOP BITS 2.

7N1

Shortcut for DATA BITS 7, PARITY OFF, and STOP BITS 1.

7N2

>Shortcut for DATA BITS 7, PARITY OFF, and STOP BITS 2.

7O1

>Shortcut for DATA BITS 7, PARITY ODD, and STOP BITS 1.

7O2

>Shortcut for DATA BITS 7, PARITY ODD, and STOP BITS 2.

7S1

>Shortcut for DATA BITS 7, PARITY SPACE, and STOP BITS 1.

7S2

>Shortcut for DATA BITS 7, PARITY SPACE, and STOP BITS 2.

8E1

>Shortcut for DATA BITS 8, PARITY EVEN, and STOP BITS 1.

8E2

>Shortcut for DATA BITS 8, PARITY EVEN, and STOP BITS 2.

8M1

>Shortcut for DATA BITS 8, PARITY MARK, and STOP BITS 1.

8M2

>Shortcut for DATA BITS 8, PARITY MARK, and STOP BITS 2.

8N1

>Shortcut for DATA BITS 8, PARITY OFF, and STOP BITS 1.

8N2

>Shortcut for DATA BITS 8, PARITY OFF, and STOP BITS 2.

8O1

>Shortcut for DATA BITS 8, PARITY ODD, and STOP BITS 1.

8O2

>Shortcut for DATA BITS 8, PARITY ODD, and STOP BITS 2.

ADD128

> Used with T160E emulation. Takes the next character entered, adds 128 to set the 8th bit, and sends the result to the host. By default, assigned to `F3` and all variations (`SHIFT` `F3`, `CTRL` `F3`, `ALT` `F3`).

ALLOW-1C/S

> Returns to the server "1" if host commands (starting with hex 1C) are allowed, or "0" if not.

ALLOW-QUIT [ON] | OFF

> If OFF, Anzio will prevent the user from quitting at the PC end. Instead, the user would have to tell the host system to logoff. This would cause the host machine to drop its end of the network or modem connection. Anzio would detect this, and then quit (unless RECONNECT is on).

> Using ALLOW-QUIT OFF can force users to terminate host-end programs properly.

ANSWERBACK *string*

> Sets the ANSWERBACK to *string*. The ANSWERBACK can be sent to the host, on host request, in certain terminal emulations. To include an ⏎ in *string*, use a vertical bar "|".

ASEND [*filename*]

> Causes Anzio to send the file *filename* to the host, using the ASCII file transfer "protocol", which consists of simply sending the file a line at a time. If no *filename* is given, you will be prompted to choose one.

AUTO-CONNECT [ON] | OFF | TOGGLE

> AUTO-CONNECT ON causes Anzio to immediately try to connect to the host when it is started up.

AUTO-LF [ON] | OFF | TOGGLE

> AUTO-LF ON causes Anzio to send a linefeed character after every carriage-return (⏎).

> A separate configuration switch, usually set from the host, determines whether to add a linefeed to any carriage-return received from the host.

```
AUX [ON] | OFF | TOGGLE
```

>   AUX ON causes Anzio to accept input from a serial device such as a barcode reader, on a communication port configured by AUX-SETUP. AUX OFF turns off that input.

```
AUX-BIDI
```

>   Puts the AUX port into bidirectional mode. Data received from the host is sent out the AUX port (instead of to the screen), and data received from the AUX port is sent to the host.

>   This mode is terminated when the host sends a sequence consisting of an Escape character followed by "[2{".

```
AUX-DEBUG [ON] | OFF | TOGGLE
```

>   Turns on or off the AUX-DEBUG feature. When this feature is on, and you do input from or output to an AUX port, such as for a serial barcode reader, a window will pop up to show data on the aux channel.

```
AUX-READ/S string
```

>   Starts reading from the AUX port, sending received data to the host, until the terminator *string* is received on the AUX port.

```
AUX-SETUP
```

>   AUX-SETUP configures a comm port (NOT the one connected to your host system) to accept serial data from an external device such as a barcode reader. You will be prompted for a port name, and then for baud rate, etc. Input will not begin until you do AUX ON.

```
AUX-WRITE string
```

>   Sends *string* to the AUX device. This does NOT alter the state of AUX input - if it was allowed before, it is allowed after this command, etc. If *string* contains spaces, use quotes around the entire *string*. This command could be used to send a trigger code to an electronic scale, for instance.

```
BACKSPACE 8 | 127
```

>   Sets the ASCII character sent for the Backspace key, either a BS (decimal 08, hex 08, ctrl-H) or a DEL (decimal 127, hex 7F).

`BASENAME/S` *filepath*

> Sends to the host (or to the nesting command, if present) the filename, and extension if it exists, of the *filepath*. That is, it strips off the disk and directory part.

`BAUD` *nnnn*

> Sets the baud rate, in bits per second, for a serial connection to the host. Available rates are determined by Windows.

`BEEP [ON] | SLOW | OFF`

> Controls the PC's beeper. BEEP ON sets the beeper to sound when a beep character (Ctrl-G, ASCII 7) is received. BEEP SLOW is the default, sounding the beep only when 1) the host issues a beep and unlocks the keyboard, and 2) there is nothing to be sent (e.g. a defined key). BEEP OFF silences the PC.

`BEEP IDLE [ON] | OFF | TOGGLE`

> BEEP IDLE ON sounds the PC beep once per minute, when the host is idle and expecting input, for example, when online and at the command level. BEEP IDLE OFF turns this feature off.

`BLOCK-HIBERNATE [ON] | OFF | TOGGLE`

> When ON, Anzio blocks the PC from going in to a low-power state, such as suspend or hibernate, because that can cause a disruption of a network connection. Note that this block can remain in effect after Anzio quits.

`BLOCK-SCREENSAVER [ON] | OFF | TOGGLE`

> When ON, Anzio blocks the screensaver from starting up. Note that this block can remain in effect after Anzio quits.

`BLOCKTYPE COLUMN | STREAM | WRAP`

> Sets the "shape" of the selection (highlight) area that is created when you drag the mouse. A COLUMN block is rectangular, and will be copied to the clipboard as several separate lines. A STREAM block is similar to that used in word processing programs, and will also be copied to the clipboard as separate lines. A WRAP block is similar to a STREAM, but lines are joined with an intervening space, except where there is a blank line.

```
BMP filename [style] [window-rect [view-rect [bright
[contrast [angle]]]]]
```

> Tells AnzioWin to open *filename*, which must be a BMP, GIF, TIF, or JPG file, and display it on the screen. The *filename* can also be a URL; Anzio will fetch the file from a web or FTP server.
>
> One bitmap at a time can be used. A bitmap can be stretched to fill the entire Anzio screen, with text over it. This is known as a WATERMARK bitmap. Or a bitmap can be repeated horizontally and vertically to fill the screen; this is a TILED bitmap.
>
> Finally, a bitmap can be displayed in a particular part of the screen, known as a *window-rect*, the corners of which can be specified in either pixel coordinates or cell coordinates. In addition to a window-rect, you can specify that only a portion of the original bitmap is displayed; this portion is called a *view-rect*. By changing the *window-rect* and the *view-rect*, various zooming, panning, and stretching effects can be created. It is also possible to alter the bitmap's *brightness* and *contrast*.
>
> It is possible in any case to have text displayed over the bitmap. Anzio will check the background color of the text to be displayed. If that background color matches the background color of the "normal" color in Anzio's Color screen, that character cell will be transparent, and the bitmap will be visible. All other background colors will block out the bitmap.
>
> If you print the screen when a bitmap is displayed, the bitmap is also printed, *unless* it is a WATERMARK or TILED bitmap.
>
> Optional parameters indicate which part of the window is affected, which part of the bitmap to use, brightness and contrast settings, and whether to rotate the bitmap before displaying it.
>
> The *style* parameter can be one of the following:
>
>> WATERMARK
>>> Specifies that the image should be stretched or shrunk to completely fill the screen, including the border areas. Overrides any *window-rect* or *view-rect* settings.
>>
>> TILED   Specifies that a bitmap smaller than the window will be repeated in the horizontal and vertical directions as necessary to fill

---

	the window. Overrides any *window-rect* or *view-rect* settings.
CELL	States that coordinates in the *window-rect* are in character rows and columns (starting at 1), rather than pixels.
PIXEL	States that coordinates in the *window-rect* are in pixels (the default).

The *window-rect* is a series of four numbers, in the sequence left, top, right, bottom, which describe the part of the screen in which to place the bitmap. If CELL is specified, these are in row and column units, otherwise they are in pixel units based on the first character cell on the screen. A value of -1 for *top* or *left* represents that edge of the screen. The *right* and *bottom* numbers may be 0, to represent the full window's right and bottom edges. The window rectangle can also be set with the BMP-WINDOW command.

The *view-rect* is a series of four numbers, in the sequence left, top, right, bottom, which describe the part of the bitmap to be displayed, in pixels. The *right* and *bottom* numbers can be 0, to represent the bitmap's right and bottom edges. The bitmap can be stretched if either *right* or *bottom* of *window-rect* is -1, or if *right* or *bottom* of both rectangles is specified. The default for *view-rect* is "0, 0, 0, 0". The view rectangle can also be set with the BMP-VIEW command.

The *bright* parameter indicates the brightness. This is a number between 0 and 2000, where 1000 is "normal". Can also be set with the BMP-ADJUST command.

The *contrast* parameter indicates the contrast at which the bitmap is displayed. This is a number between 0 and 2000, where 1000 is "normal". Can also be set with the BMP-ADJUST command.

The *angle* parameter specifies the bitmap's rotation, in degrees.

See section 9.3 *Variable Substitution*.

BMP-ADJUST *bright* [*contrast*]

> Sets the background bitmap's brightness and contrast, as described above. If your video adapter is running in 256-color mode, Anzio will manipulate the palette in order to change brightness and contrast. In true-color modes, Anzio will internally adjust the bitmap, which will take longer.

BMP-ANGLE *degrees*

> Changes the angle of rotation for the screen bitmap.

BMP-CLOSE

> Closes the current background bitmap, if any, and removes it from the screen. Use the BMP command or the File/Background Bitmap menu item to set the background bitmap.

BMP-VIEW *left top right bottom*

> Sets the background bitmap's view rectangle as described under the BMP command.

BMP-WINDOW [*style*] *left top right bottom*

> Sets the background bitmap's window rectangle as described under the BMP command.

BOX *c1 c2 r1 r2* [type]

> This command draws a box on the screen from character position (*c1*, *r1*) to (*c2*, *r2*). The optional *type* is the line style, either SINGLE (default) or DOUBLE.

> The corner offsets may be either absolute (from the upper-left screen character) or cursor-relative. For example, BOX .-1 .-5 10 20 indicates a box from one character to the left of the cursor (.-1) and five up (.-5) to screen position 10,20.

BREAK

> Sends a BREAK signal to the host computer.

C332-NEXT
C332-PREV

> Goes to the next or previous session when using the C332's multi-session capability. These commands are usually included as macros attached to special keys.

```
C332-SEND [filename]
```

Uploads the indicated file to the host, using the C332 protocol. If *filename* is not given, you will be prompted to select a file.

```
CALC
```

Starts a simple four-function calculator on the status line. The calculator can work in decimal or hex mode.

The calculator has several commands:

+ - * /	Addition, subtraction (minus), multiplication, division.
⏎	Functions as a '+=' key.
DEL	Clears the last entry. If pressed twice, clears the total. '
ESC	Exits the calculator, leaving the result on the status line.
H	Switches between decimal (default) and hexadecimal mode (indicated by 'HEX' on the status line). Changing modes converts the current result to the new base.
F1	sends the result to the host in the current mode and exits calculator mode. The result will remain displayed for 5 seconds.

```
CALL macro
```

This command allows one macro to start another macro. When the second macro finishes, the first (calling) macro will resume. The *macro* shown above can be either the actual keystroke, achieved by inserting the PG DN key when entering the command. Or, it can be the *name* of the key, as reported in the KEYS display, such as "PgDn".

See also INVOKE.

```
CALLDIGITS digits
```

After an initial DIALCALL, this command will send additional digits.

```
CAPTURE [ON] | OFF
CAPTURE filename | CLOSE
CAPTURE WPRN
```

> The CAPTURE feature saves a copy of all displayable data received from the host. You cannot capture data and download a file at the same time, since the receive buffer is also used for file transfer.

> The way that capture deals with control characters varies, depending on the settings in Advanced Options.

> If an output file is open (from the OPENO command), captured data is written to that file. Use CAPTURE ON and CAPTURE OFF to start and stop recording.

> CAPTURE *filename* opens a file and begins capture. CAPTURE CLOSE turns off capture and closes the file.

> CAPTURE WPRN sends the captured data to the Windows printer driver.

```
CD [unit:]directory
```

> Sets the default ("logged") disk drive and directory. The logged disk and directory are used for local file operations (DIR, OPENI, etc.) when no drive or directory is specified.

> The optional disk unit must be followed by a colon.

> Synonymous with the LOG command.

```
CHARSET name
CHARSET/S name
```

> Specifies which character set coding is used by the host system, which in turn causes Anzio to translate characters appropriately.

> *Name* can be a language, representing a National Replacement Character (NRC) set.   If *name* is INTERNATIONAL, no NRC translation is done. Possible NRC names are: CANADIAN, FINNISH, FRENCH, GERMAN, ITALIAN, NORWEGIAN, SPANISH, SWEDISH, SWISS, UK, SLOVENIAN, and JAPANESE.

> *Name* can be also be a specific character set name. Character set names can be used only with ANSI emulations (VT-*xxx*, SCOANSI, AT386, LINUX, and ANZIO).

ISO	Standard ANSI (Windows) character set. The actual character set is dependent on the Windows installation.
OEM	The DOS or "linedraw" character set. The actual character set is dependent on the Windows installation.
ISO-8859-*x*	*x* is 2 through 9 or 15
DEC	Slightly different from standard ISO
CP-*xxx*	*xxx* is the number of a Windows or DOS *character codepage*, such as 850. Requires Windows support for the particular codepage.
T160E	Innovative Interface's T160E multilingual coding scheme
CCCII	Chinese, Japanese, and Korean combined character set (also known as EACC)
ASMO449	Arabic
KOI8-R	Russian
VISCII	Vietnamese
UTF8	Byte-encoded Unicode characters
LINUX-UTF8	Byte-encoded Unicode characters, as used by a Linux console
USMARC	US Library of Congress scheme
CUSTOM	Allows use of a user-editable translation file, CUSTOM.UNI
other	Other encodings as provided by additional *.UNI files in Anzio's directory

CHARSET/S returns a status result to the host: "00" for success, and "01" for failure. That way a host program can determine whether this installation of Anzio has support for a particular encoding.

See also PASSTHRU-CP to set the printer code page.

```
CHOOSEPRINTER [name | DEFAULT]
CHOOSEPRINTER/S
```

CHOOSEPRINTER brings up the Windows Printer Setup dialog box, so the user can select or set up a printer.

The optional *name* is the name of a Windows printer, exactly as it appears in the Windows Printers box. The *name* is the entire remaining part of the command, even if it contains spaces; quotes are optional (unlike normal syntax rules).

Use CHOOSEPRINTER DEFAULT to choose the current default Windows printer.

CHOOSEPRINTER/S will return to the host the Windows name of the currently selected printer, or "DEFAULT" id none is established yet.

CLIP [*c1 c2 r1 r2*]

Copies the screen text (or just the given region) to the Windows clipboard. If the columns and rows are not given, Anzio will copy the entire screen to the clipboard, unless a highlight is in place, in which case it will copy just the highlighted area. However, if the configuration file's "copy-full-screen" line is set to "0", and no highlight is defined, Anzio will not copy anything to the clipboard.

If the coordinates are given, they can be cursor-relative – see BOX.

Anzio stores the screen in the clipboard in several ways: ISO text, OEM text, Unicode text, Rich Text Format (RTF), and bitmap. However, copying as RTF can be turned off, in Edit/Advanced Options/Mouse.

When you paste into another program, such as Microsoft Word, you may be given options of "Paste" and "Paste Special". "Paste" tells the program to choose its preferred format of those available, such as RTF. If you prefer to paste in a different format, such as plain text or bitmap, choose "Paste Special"; the program will then let you pick from the available formats.

Use PASTE in Anzio to send the clipboard text to the host.

CLIP "*string*"

Sends the literal *string* to the clipboard. The quotes are necessary to differentiate this from the other form of CLIP.

CLONE [/NOSAVE] [*command*]

Creates another Anzio session, using the same parameters as the current session. That is, it clones the connection. If the parameters have been changed, you will be prompted to save them first, unless /NOSAVE has been specified.

If *command* is specified, this initial command will be sent to the server by the new session as soon as authentication has taken place, by means of the "/I" command line parameter. To get a terminating <enter>, include a pipe symbol.

CLOSEI
CLOSEI/S

Closes the input file, if any. See OPENI.

CLOSEI/S also sends a result code to the host:

	00	Completed successfully
	01	Error occurred

CLOSEO
CLOSEO/S

Closes the output file. See OPENO.

CLOSEO/S also sends a result code to the host:

	00	Completed successfully
	01	Error occurred

COLOR

Starts a dialog for specifying which colors/attributes are used for normal text, highlights, etc. Depending on the terminal being emulated, there are up to four video attributes: reverse video, bold or half intensity, underline, and blink. These may be used in any combination, for 16 possibilities.

Anzio has an internal translation table to change each combination of emulated attributes into a character-level parameter for the PC's video driver. The COLOR dialog is used to change that translation table.

Press the key for the attribute combination to change (0 through 9, A through F), then enter the two-byte color/attribute code from the table shown at the right.

The P key (for *palette*) cycles through several combinations of blink/underline and bright/dim attributes.

Press ESC to leave the COLOR dialog without changes, or ⏎ to save the changes.

You can also reset all colors to our defaults, as indicated.

See COLOREDIT for an easier way to do this.

COLOREDIT

> Starts a dialog for specifying which colors/attributes are used for normal text, highlights, etc. Depending on the terminal being emulated, there are up to four video attributes: reverse video, bold or half intensity, underline, and blink. These may be used in any combination, for 16 possibilities.

> Anzio has an internal translation table to change each combination of emulated attributes into a character-level parameter for the PC's video driver. The COLOR dialog is used to change that translation table.

> The "Attribute type" setting determines the amount of attribute data that can be stored, and may need to be set a certain way for particular terminal types. This is sometimes referred to as the "palette".

> When "Full attributes" is checked, Anzio will track and have available 16 foreground colors, 16 background colors, blink, and underline.

> When "8 bit attributes" is checked, Anzio emulates (with some variations) the approach of some color terminal types (such as SCOANSI) that were based on PC hardware, where only 8 bits of attribute are available. This provides 16 background colors (4 bits) and 8 background colors (3 bits). The use of the 8th bit is controlled by the "Blink/Underline" setting. It can be used for Blink or Underline; or it can be used to control background brightness.

> If Blink/underline is set to Off, then 16 background colors are available.

> If Blink/underline is set to "Blink", blink and 8 background colors are available. You can choose a bright set or a normal set with the "Bright backgrounds" checkbox.

> If Blink/underline is set to "Underline", underline and 8 background colors are available. You can choose a bright set or a normal set with the "Bright backgrounds" checkbox.

> To choose a standard set of colored attributes, click "Use color defaults". To use a standard set of monochrome attributes, click "Use monochrome defaults".

To change the treatment of a particular attribute combination, click on the "X" next to its name. For instance, if the server sends a sequence for "bold reverse", and you want to make that text display with a particular color combination, click the "X" next to "Bold reverse". Another dialog will pop up to let you choose the colors for the foreground (text) and background for bold reverse text. Depending on your palette setting, you may also be able to choose blinking and/or underlining. Click OK.

To change how a particular color name, such as "brown", displays, click "Change color map", and then click the color sample next to "Brown". This will bring up a Windows standard color selection dialog, where you can adjust the characteristics of "brown", even making it a totally different color.

When you OK out of all the color settings, some changes will not take effect until you receive new text from the server.

COMMTYPE *name*

Tells Anzio which communication method it should use to communicate with the host system. Changing the COMMTYPE causes an existing connection to be dropped.

Possible values for *name* are

> SERIAL
> TELNET or WINSOCK
> RLOGIN
> MODEM
> SSH

COMPOSE/S *x*[*y*]

Composes a special character, usually a combination of a base character and a diacritic, as represented by the two-character sequence *xy*. If *y* is not given, the user will be prompted for it. For instance,

> COMPOSE/S :A

will generate an A-umlaut, and send it to the host.

The first character in the sequence usually represents a diacritic, as follows:

Character	Meaning
`	grave
'	acute
^	circumflex
>	circumflex
~	tilde
-	macron
(	breve
*	dot above
:	umlaut
"	umlaut
?	hook above
<	hacek
,	cedilla
.	dot below
_	underline
=	double underline

All standard Unicode character/diacritic combinations using the above diacritics (as the first character) are supported. In addition, Anzio defines the following list of sequences and the generated characters:

`a	à		'd	ð
`e	è		'D	Ð
`i	ì		-D	Ð
`o	ò		'Y	Ý
`u	ù		'y	ý
`A	À		^a	â
`E	È		^e	Ê
`I	Ì		^i	Î
`O	Ò		^o	Ô
`U	Ù		^u	Û
'a	á		^A	Â
'e	é		^E	Ê
'I	í		^I	Î
'o	ó		^O	Ô
'u	ú		^U	Û
'A	Á		:a	Ä
'E	É		"a	Ä
'I	Í		:e	Ë
'O	Ó		"e	Ë
'U	Ú		:i	Ï

"¡	Ï		&o	œ
:o	Ö		Oe	œ
"o	ö		&O	Œ
:u	ü		OE	Œ
"u	ü		/o	ø
:A	Ä		o/	ø
"A	Ä		/O	Ø
:E	Ë		O/	Ø
"E	Ë		C/	¢
:I	Ï		C|	¢
"I	Ï		!!	¡
:O	Ö		??	¿
"O	Ö		L-	£
:U	Ü		L=	£
"U	Ü		Y-	¥
:y	ÿ		Y=	¥
"y	ÿ		SO	§
:Y	Ÿ		S0	§
"Y	Ÿ		S!	§
:<space>	¨		XO	¤
"<space>	¨		X0	¤
,c	ç		CO	©
c,	ç		C0	©
,C	Ç		A_	ª
C,	Ç		<<	«
,<space>	¸		0^	°
~a	ã		O^	°
~n	ñ		+-	±
~o	õ		2^	²
~A	Ã		3^	³
~N	Ñ		/U	µ
~O	Õ		P!	¶
&a	æ		.^	·
Ae	æ		1^	¹
&s	ß		O_	º
Ss	ß		>>	»
*A	Å		14	¼
Th	Þ		12	½
*a	å		RO	®
&A	Æ		_^	‾
AE	Æ		34	¾

-:	÷
-d	Đ
TH	Þ

TM	™
e=	€

```
CONNECT
[commtype://][username[:password]@]hostname[:port]
```

> Causes Anzio to drop its current connection (if it has one), and establish a new connection. If just CONNECT is done, Anzio will reconnect according to all its current settings. Or, the command can specify a new *hostname*, a new *commtype*, etc. See the COMMTYPE command for possible values for *commtype*. For instance, the following command will establish a username of "sally" and establish a modem connection to the indicated phone number:

> ```
> CONNECT modem://sally@555-4321
> ```

> See section 9.3 *Variable Substitution*.

```
COPY filename newfile
COPY/S filename newfile
```

> Copies *filename* to *newfile*. If *newfile* exists, it is overwritten without warning.

> COPY/S also sends a result code to the host:

> | 00 | Completed successfully |
> |----|------------------------|
> | 01 | Error occurred |

> If *newfile* is "WPRN", the file identified by *filename* is sent to Anzio's printer driver.

> If *filename* and/or *newfile* contains the string "://", that name is interpreted as a SUPERFILE name.

> See section 9.3 *Variable Substitution*.

```
CURSOR [BLINK] [ON] | OFF
```

> Tells Anzio whether you want the standard blinking cursor, or the non-blinking pseudo-cursor.

```
CURSOR BLOCK | UNDERLINE
```

> Tells Anzio whether you want a block cursor or an underline cursor.

```
DATA [BITS] 5 | 6 | 7 | 8
```

>   Sets the number of data bits in the serial communication
>   protocol, usually 7. Note that this bit count does not include the
>   parity bit, if any.

```
DATE/S [format] [offset]
```

>   Sends today's date, as known by the Windows clock, to the host.
>   If an *offset* is given, that number of days is added to (subtracted
>   from if negative) the current date.

>   The default format is "mm/dd/yy". If *format* is given, it is a string
>   of tokens representing the elements of the date to be sent, as
>   follows:

d	Displays the day as a number without a leading zero (1-31).
dd	Displays the day as a number with a leading zero (01-31).
ddd	Displays the day as an abbreviation (Sun-Sat).
dddd	Displays the day as a full name (Sunday-Saturday).
m	Displays the month as a number without a leading zero (1-12).
mm	Displays the month as a number with a leading zero (01-12).
mmm	Displays the month as an abbreviation (Jan-Dec).
mmmm	Displays the month as a full name (January-December).
yy	Displays the year as a two-digit number (00-99).
yyyy	Displays the year as a four-digit number (0000-9999).

>   Note that day and month names and abbreviations will vary with
>   the Windows locale.

```
DDEEXECUTE command
```

>   Executes a DDE *command*. A DDE conversation must have
>   been established with another program, with the DDEINITIATE
>   command.

DDEINITIATE *program topic*

>   Causes Anzio to act as a DDE client, and begin a DDE
>   conversation with *program*, about *topic*. Only one DDE
>   conversation can be in effect at one time. It is terminated with a
>   DDETERMINATE command. Note that *program* is a special
>   name used for DDE by another program; it might NOT be the
>   same as the executable's name.

DDEPOKE *item data*

>   Sends data into another program, using DDE. A DDE
>   conversation must have been established with another program,
>   with the DDEINITIATE command.

DDEREQUEST/S *item*

>   Requests data from another program using DDE. A DDE
>   conversation must have been established with another program,
>   with the DDEINITIATE command. The data returned is sent by
>   Anzio to the host.

DDETERMINATE

>   Terminates a DDE conversation started with DDEINITIATE.

DEBUGKEYS [ON]|OFF

>   Turns on or off a window the displays information about
>   keystrokes received from Windows or third-party applications.

DEFAULTS

>   Saves the current parameter settings to a file.

DEFINE *x text*

>   Defines macro *x* as *text*. See *Defined Keys and Macros*, page
>   45.

DELAY *n*
DELAY/S

>   DELAY sets the time delay between characters sent to the host.
>   This is necessary on some systems to prevent overloading the
>   communication hardware on the host. *n* is a number from 0 to
>   65535 in units of 10 microseconds.

>   DELAY/S sends the current DELAY and LINE DELAY settings to
>   the host.

See also LINE DELAY.

```
DELETE filename
DELETE/N filename
DELETE/S filename
```

Deletes a PC file. Anzio will display an error message if the file does not exist.

DELETE/N does not display an error message if the file is not found.

DELETE/S also sends a result code to the host:

> 00   Completed successfully
> 01   Error occurred

```
DIAL phone_number [wait retries]
```

Tells the modem to call *phone_number* to initiate a new connection.

The recommended way for Anzio to communicate with a modem is with the communication type set to "modem". However, for historical reasons, it is still possible to communicate with a modem (although not some newer modems) as serial devices. For that approach, Anzio allows you to specify additional parameters, so that if the connection is not successful on the first try, it will keep trying *retries* times, waiting *wait* tenths of a second between retries.

```
DIALCALL
DIALCALL "phone-number"
```

Used for dialing a voice call. Both forms of the command will drop (hang up) an existing call, if there is one. Then, if a *phone-number* is given, it will be dialed. This should be in so-called canonical form: "*+country (areacode) phonenumber*" Be sure to include the quotes.

```
DIR [pathname]
DIR/S [pathname]
```

Displays all file names in the current LOG directory, or the *pathname* directory if specified, using a standard Windows dialog.

DIR/S also sends the selected file name either to the host or to another function. See *DIR/S Example*, page 56.

```
DIR-TERMINATOR string
```

> Specifies what string Anzio should use to indicate when DIR/S, FIND/S, FINDNEXT/S, or FILEINFO/S fails or terminates. The default is "\\", but that is difficult to deal with in a Unix environment.

```
DIROUT/S desc ext [caption]
```

> Displays a standard Windows "write file" dialog box, prompting the user to select a filename to create, with a description *desc* and a file extension *ext*. Optionally, the *caption* for the dialog box can be specified.

> The resulting filename will be returned to the host.

```
DOWNLOAD-DIR [pathname]
```

> Sets *pathname* as the default directory used for downloads and file captures.

> If *pathname* is not specified, cancels any previous DOWNLOAD-DIR or DOWNLOAD-LOCK.

> DOWNLOAD-DIR does not prevent files from being created explicitly in other directories. See DOWNLOAD-LOCK.

```
DOWNLOAD-LOCK [pathname]
```

> Forces all downloads and captures to be in *pathname*.

> If *pathname* is not specified, cancels any previous DOWNLOAD-DIR or DOWNLOAD-LOCK.

```
DROPCALL
```

> If a voice call is in effect, it will be terminated.

```
EJECT
```

> Ejects a page from the printer and closes the print job. See also FLUSH.

```
EJECTMEDIA drive[:]
```

> Tells Windows to make the indicated "disk drive" unavailable. This might be a CD drive, for instance, in which case the drawer will open. Or it might be a "thumb drive" or any other removable medium.

```
EMAILSETUP
EMAILSETUP vbl=value [vbl=value...]
```

>Configures Anzio to send emails.

>EMAILSETUP by itself will prompt the user to fill in various parameters.

>If one or more variable/value pairs is specified in the form *vbl=value*, each variable will be set and no user interaction will occur. Possible variables are:

TYPE	Specifies the email method to be used. Options are SMTP, CDO, MAPI, and OUTLOOK.
SERVER	The email server's name or IP address.
PORT	The email server's port number.
USER	The username for access to the mail server.
AUTH	Set to ON or TRUE if the server requires authentication; OFF or FALSE otherwise.
PASSWORD	The password to use for authentication.
FROM	The name of the sender that should appear in emails.
FROMADDRESS	The email address of the sender that should appear in emails.
PROFILE	The profile name where required, such as for MAPI.
HTML	Set to ON or TRUE to send emails as HTML; OFF or FALSE to send as plain text.
CONFIRM	Set to ON or TRUE to prompt user to complete and confirm an outgoing

email message; OFF or
FALSE otherwise.

`END [/NOSAVE] [`*`returncode`*`]`

Stops Anzio. Same as STOP. Can be abbreviated as E.

If /NOSAVE is specified, settings will not be saved to the settings
file. Otherwise, behavior will be as set in Edit:Advanced
Options:Security:Prompt to save settings.

If a numeric *returncode* is specified, that value will be returned by
the Anzio session to whatever process initiated it. Otherwise,
zero will be returned.

`ENV/S `*`variable`*

Sends the *value* of the environment *variable* to the host,
terminated by a RETURN. Or, this variable could be an internal
Anzio environment variable — see 9.3 *Variable Substitution*.
Note that in this case the *variable* is NOT surrounded by
brackets.

`ENVIRONMENT `*`string`*

Sets Anzio's internal ENVIRONMENT variable to *string*. This
string is sent to the host system at the beginning of a telnet or
SSH session. Typically used with UNIX systems. Note that
some telnet daemons or SSH daemons will pass on only a
limited set of environment variables, such as DISPLAY,
LPDEST, and/or PRINTER.

string is a list of environment variables and their values,
separated by semicolons. The format for each variable is:

> *`name`***`=`***`value`*

where *value* is a number, string, or nothing (null), or

> *`name`***`=$`**`{`*`variable`*`}`

where *variable* is a variable name which gets translated as
explained in 9.3 *Variable Substitution*.

➢ The curly-brace ("{") character has a special meaning inside a macro.
So if you wish to include an environment variable inside a macro, it
must be in form:
$~{value}

```
EVENT eventname macro [param]
```

> The EVENT command tells Anzio to execute a certain macro when a certain event occurs. The *macro* can be an alphanumeric key, or the name of the key, such as "F10". The *param* is used only where indicated below. The *eventname* can be one of the following:

ONSTARTUP	Executes when Anzio starts up. This is equivalent to the STARTUP-MACRO command.
ONCONNECT	Executes when a successful connection is established with the server. This overrides the Login Wizard, so Login Wizard does not run.
ONLOGIN	Executes when a successful login has been completed, via either SSH or the Login Wizard.
ONWAKEUP	Executes when the PC comes out of a low-power state such as hibernation. This can occur when you open up a laptop, for instance.
ONERROR	Executes when an error occurs in command processing. Equivalent to the ONERROR command.
ONDROP	Executes when the user does a drag-and-drop operation and drops a filename on the Anzio window. The variable ANZ_DROP_NAME will contain the filename dropped. The variables ANZ_DROP_X and ANZ_DROP_Y will contain

		the x and y coordinates (in character units, based on 1) where the drop occurred.
	ONIDLE	Executes when there has been no user activity or data from the host for a certain number of seconds. If *param* is given, that is the number of seconds (which can have a decimal part); otherwise the last specified value is used.
	ONBEEP	Executes when Anzio receives a "bell" signal (hex 07, decimal 07, control-G) from the server.
	ONQUIT	Executes when the user attempts to quit from Anzio locally, such as via the File/Quit menu item or clicking the close ("X") box on the title bar.
	ONHOSTPRINT	Executes when a passthrough print job begins.
	ONDIALIN	Executes when the modem detects a ring.
	ONCALLERID	Executes when a call containing caller ID comes in. Takes precedence over ONDIALIN.

The *macro* parameter can be an alphanumeric key, or it can be the key name, such as "sf10" for shift-F10.

EXPECT *value1 value2*

Anzio will evaluate *value1* and *value2*, expecting them to be equal strings. If they are not, an error condition exists; if an ONERROR macro is in effect, it will be executed, otherwise a message will be displayed. Note that either value might be derived from an embedded function.

FAXSETUP

>   Prompts the user for various parameters for faxing the results of a print job.

FILECLEANUP *wildcard days*

>   Searches for files whose names match a *wildcard* pattern, that are at least *days* old, and deletes them. This can be used to clean up downloaded files, for instance.

>   USE WITH CAUTION!

FILEINFO/S *filename*

>   Allows a host system to determine whether a particular file currently exists on the PC, and if so, some of its characteristics.

>   Returns to the host an ASCII string containing information on the file named *filename*. If *filename* is a wildcard, it returns information on the first matching file. If no file is found, Anzio returns the DIR-TERMINATOR string, which by default is two backslashes ("\\").

>   The information returned on an existing file is in four fields separated by commas, representing:

>   a) File size, in decimal

>   b) The file's attribute flags, as defined by DOS

>   c) The date/time of last change, in the local time zone, as *yyyymmddhhmmss*

>   d) The date/time of the last change, in UTC (Greenwich) time, in the same format.

FILL *c1 c2 r1 r2* [*char*]

>   Fills a rectangular area of the screen with spaces, or the optional character *char*. The offsets may be absolute or cursor-relative, for example, FILL 1 1 .+5 .+5.

FIND/S *filespec*

>   Finds the first file or directory name matching the name or wildcard entry in *filespec* and sends it to the host. If it is a directory, Anzio will add a backslash ("\") to the end.

If there are no matching files, Anzio sends the DIR-TERMINATOR string, which by default is two backslashes ("\\"). However, if an ONERROR macro is in effect, that will be triggered instead. See also FINDNEXT/S.

Note that FIND/S can be used to by a host system to determine if a particular file name exists on the PC. But see also FILEINFO/S.

FINDNEXT/S

Finds the next file or directory name matching the wildcard entry in the previous FIND/S command, then sends the name to the host.   If it is a directory, Anzio will add a backslash ("\") to the end.

If there are no more matching files, Anzio sends the DIR-TERMINATOR string, which by default is two backslashes ("\\"). However, if an ONERROR macro is in effect, that will be triggered instead. See also FINDNEXT/S.

FLASH [*n*]

Causes Anzio's entry on the task bar (at the bottom of the screen) to flash *n* times (default is 3). This is typically done when Anzio is not in focus, and is a way of indicating to the user that Anzio needs attention.

FLUSH

Sends any remaining data in the print buffer to the printer, and closes the print job. If spooling is enabled on the printer (network or Windows local), this will release the print job for printing. See also FLUSHTIMER.

This command will eject the page only if PRINT LEVEL is High. See also EJECT.

FLUSHTIMER *time*

Sets the number of seconds between FLUSHes (default 5 seconds). After *time* seconds have elapsed since Anzio sent something to the printer, Anzio calls the FLUSH command to allow the data to be printed.

To disable this feature, set *time* to 0. For example, you may need to print two screen dumps per page, or your passthrough print data may take longer than *time* to arrive.

```
FONT [name] [size] | LARGER | SMALLER | BOLD
```

> Sets the displayed Windows screen font name and size. Anzio will change the window size to follow the font size, unless the window is zoomed.

name	Font name, enclosed in quotes if it contains spaces, such as "Courier New".
size	Font size, as either *height* or *height x width*.
LARGER	The next larger font size, for example, from 10 to 12.
SMALLER	The next smaller font size.
BOLD	Selects a bold font.

> Note that your PC may not have the indicated font and size.

```
FONT-ALT fontnames
```

> Tells Anzio what alternate fonts should be used for the screen, to display characters not found in the primary font. The *fontnames* can be a list of fonts by their Windows names, separated by commas. Fonts will be tested in the order indicated. To tell Anzio NOT to use a particular font, put a minus sign in front of its name. The last entry should generally be "auto", telling Anzio to look through any other fonts on the Windows system. The default is simply "auto".

> Because many font names contain spaces, you will probably need to have the entire *fontnames* string in double quotes.

```
FTP [PASSIVE]
FTP NEW
```

> These commands initiate an *interactive* File Transfer Protocol (FTP) session over a network connection. They require WININET.DLL, which comes with Internet Explorer and some versions of Windows. The dialog box provides an easy point-and-click interface that allows you to move files easily between the PC and a host system, which might or might not be the same host as you are telnetting to.

> The command

```
        FTP
```

---

will initiate an FTP connection to the same host, the same username, with the same password as you are currently telnetting. If you include the word "PASSIVE", then the connection will be established in FTP passive (PASV) mode, which might be required by a firewall at your site.

The command

```
FTP NEW
```

will prompt you first for the hostname, port, username, and password for the connection. It also allows you to indicate that you need a passive (PASV) connection, via a check box.

```
FTPDELETE hostfile
FTPDELETE/S hostfile
```

Causes Anzio to use FTP commands to delete a file on an FTP server.

hostfile can be a name, a pathname (with either kind of slash), or a URL matching:

username[:*password*]@*hostname*[:*port*]/*filepath*

(note that you can't do *username* and *filepath* but not *hostname*).

The "/S" causes a status result to be sent to the host ("00" means success).

```
FTPGET [ASCII] [PASSIVE] [hostfile] [pcfile]
FTPGET/S [ASCII] [PASSIVE] [hostfile] [pcfile]
FTPPUT [ASCII] [PASSIVE] [hostfile] [pcfile]
FTPPUT/S [ASCII] [PASSIVE] [hostfile] [pcfile]
```

These commands do *programmed* File Transfer Protocol (FTP) transfers over a network connection. They require WININET.DLL, which comes with Internet Explorer and some versions of Windows. They allow AnzioWin to work as an FTP client. However, these commands **can** be issued from the host (through Anzio's usual command protocol), which effectively allows the host to control the FTP process.

The ASCII modifier causes an ASCII (text mode) transfer, which converts end-of-line delimiters.

The PASSIVE modifier causes passive FTP protocol to be used.

These commands assume that the host name, username, and password for the FTP session are the same as for the Anzio telnet or SSH connection. Thus the command can be as simple as

```
FTPGET hostfile
```

hostfile can be a name, a pathname (with either kind of slash), or a URL matching:

username[:*password*]@*hostname*[:*port*]/*filepath*

(note that you can't do username and filepath but not hostname).

You can enter an empty *hostfile* by using a pair of double quotes: "". Then the hostfile name will be assumed to be the name part (without the path) of the *pcfile*.

If neither *hostfile* nor *pcfile* is not specified, you will be prompted to select a PC filename, and the host file name will be assumed to be the same (without the path).

If *pcfile* is not given, the PC file name will be derived from the basename (name without directories) of the *hostname*.

The "/S" causes a status result to be sent to the host ("00" means success). Anzio's DOWNLOAD DIRECTORY will affect downloads (FTPGET[/S]).

```
FULL [DUP]
HALF [DUP]
```

Sets the duplex mode:

> FULL DUP   Anzio expects the host to echo back each keystroke to the screen.
>
> HALF DUP   Anzio displays each keystroke sent to the host.

If you are getting double characters, you need to set FULL DUP. If you are not getting characters on the screen at all, you need to set HALF DUP.

```
GAUGE [ON] | OFF | TOGGLE
```

Controls the gauge line on the bottom of the screen. The gauge line displays '....|..*.1.*..|....2...' to help you locate columns. The gauge line also shows the cursor line and position.

```
GETCOOKIE/S url [name]
```

> Returns to the host the value of an Internet cookie, such as might have been left by a web page (or some other program). Cookies can be identified by *url* and optionally by *name*. If this command does not specify a *name*, the entire cookie for *url* will be returned.

```
GETPARAM/S [section]
GETPARAM/S [section]variable
```

> ➢ In this command the square brackets are part of the command, and are not used to indicate optional content.

> Returns to the host a particular variable, or an entire section, from Anzio's current working parameters, in the same form that it would be stored in Anzio's settings file. A host process could later reset this value with a SETPARAM.

```
HALF [DUP]
```

> Sets Half Duplex mode. See FULL DUP.

```
HANGUP
```

> Hangs up and terminates a modem connection.

```
HELP
```

> The HELP command displays an old DOS style help screen with many available commands, any open file names, and the current option settings. Press ⏎ to exit the HELP screen and return to your "working" screen.

> In the HELP screen, you can enter a command without pressing F6 . After this command is processed, either the HELP screen is redisplayed, or you will return to the working screen.

> Note that the PRINT command will print the HELP screen, not the working screen.

```
HEXPATCH address bytes
```

> No longer used.

```
HIGHLIGHT x1 x2 y1 y2 [COLUMN | STREAM | WRAP]
```

> Sets the on-screen highlight to the part of the screen indicated. The type of block can optionally be specified. Note that this command can be created easily by recording a macro.

```
HOLD [ON] | OFF | TOGGLE
```

> Suspends (holds) output from the host to the screen. The HOLD
> TOGGLE command can be attached to a macro key. See also
> SCROLL-LOCK.

```
HOSTNAME/S
```

> Causes Anzio to send its current host name to the host. The
> host name can come from a command line parameter, a telnet or
> SSH address, or from the network parameters.

```
IGNULL [ON] | OFF | TOGGLE
```

> Tells Anzio whether to ignore nulls (hex 00) in the incoming data
> stream. Should be IGNULL OFF unless instructed otherwise by
> Rasmussen Software.

```
IMAGEBOX
IMAGEBOX filename [CELL | SCREEN] [FLOAT] x1 y1 x2 y2
[options]
```

> Controls an image viewer window, referred to as the image box.
> IMAGEBOX (by itself) closes an existing image box. Other
> forms of the command are described below.

> ➢ This is different from the background bitmap controlled by the BMP
> commands. It is also different from the movie window controlled by
> the PLAYMEDIA command.

> One image box at a time can be open. It can be positioned over
> part of the main Anzio window, or it can be a "floating" window,
> that the user can move and resize. It can be on a second
> monitor. If "Track window position" is on (in the View menu), the
> image box's position will be tracked between sessions, if settings
> are saved.

> The image box can display files in these formats: BMP, GIF,
> JPG, TIF, PCX, DCX, image PDF (but not "searchable" PDF),
> and PCL. TIF, PCL, and PDF files can contain multiple pages.
> The *filename* can be a URL, specifying that the file will come
> from an FTP, HTTP, or HTTPS server. Or, the *filename* can be
> "scan://" (optionally with additional parameters) to cause Anzio to
> scan and view a document. Scans can also be multiple pages.

The parameters *x1 y1 x2 y2* define the left, top, right, and bottom of the rectangle in which the image will appear. By default, these are in pixels, relative to the top left corner of the client area of AnzioWin, below the toolbar if displayed. If CELL is stated, they are in units of character cells, where "1" is the top (left) character position. If SCREEN is stated, they are in units of pixels relative to the top left edge of the video screen (desktop). If FLOAT is stated, the image window is made moveable and sizeable, by wrapping it with a suitable Windows frame, while keeping the client area in the same place.

It is possible to specify a negative *x1*, etc.

The *options* available include:

> REFRESH=*num*
>> Causes AnzioWin to reload the image by rereading *filename* every *num* seconds (which can have decimals). This can reread a file written by a web camera, for instance.

> ASPECT=FIXED | VARIABLE
>> Specifies whether the aspect ratio of the image is maintained (fixed), which is the default. VARIABLE allows the user to distort the image.

> PAGE=*n*   Specifies which page of a multi-page file will be shown in the window.

```
INK [ON] | OFF | TOGGLE
```

Turns on or off the ability to mark up the main window. Markup is possible using the mouse (on most Windows XP systems), or the stylus on a TabletPC.

Markup allows you to draw on the main Anzio window, as well as erase your marks. An alternative popup menu is available in ink mode; it lets you change your ink color, your eraser configuration, etc.

If you print the screen while ink is active, the ink markups will print also. If you copy to the clipboard, the ink markups will be part of the bitmap image put into the clipboard. In a word processor, you can do Paste Special and select a bitmap format, and you will get a screen shot with markup. This can be useful for creating user documentation.

```
INK CLEAR
```

Clears (erases) any ink markup on the main screen.

```
INTERACTIVE-PRINT [ON] | OFF | TOGGLE
```

If INTERACTIVE-PRINT is on, then when any passthrough print job is received from the server, Anzio will open the Printer Setup dialog to allow you to configure your print output. If OFF, print jobs will be handled as previously configured, with no user interaction.

```
INTERPRET
```

Displays the last 8192 characters received from the host. Non-printable characters appear as reverse-video hex numbers. This display is useful when something strange has occurred at the terminal, since it shows the input conditions.

You can view high non-ASCII characters (between hex A0 and FF) as characters in the screen's character set, as 7-bit characters, or as hex codes. Use the M key (for *mask*) to cycle through these view options.

Use PRINT to print the display.

```
INVOKE macro
```

Starts the indicated *macro*. If used inside a macro, the calling macro stops and the invoked macro continues. The *macro* shown above can be either the actual keystroke, achieved by inserting the PG DN key, for instance, when entering the command. Or, it can be the *name* of the key, as reported in the KEYS display, such as "PgDn".

See also CALL.

```
JUMP OFF | MEDIUM | FAST
```

Tells Anzio how to scroll the screen for rapid incoming data:

OFF	Shows every line.
MEDIUM	Repaints the screen when the display is a half-screenful behind.
FAST	Repaints only when the display is a complete screenful behind.

```
KCOMMAND command [parameters]
```

Sends a Kermit *command* to a host. The results of the *command* will depend on the host Kermit implementation. Anzio supports the following commands:

```
KERMIT command
CWD newdir
DIRECTORY spec
ERASE spec
FINISH spec
HELP spec
LOGIN spec
JOURNAL spec
COPY file1 file2
LOGOUT spec
MESSAGE spec
PROGRAM spec
QUERY
RENAME file1 file2
TYPE filename
USAGE param
VARIABLE
WHO
```

```
KEEP   [F7|c1 c2 r1 r2]
KEEP/N [F7|c1 c2 r1 r2]
```

KEEP sends the screen contents to the output file opened by OPENO. If you OPENO LST:, the printer will be your output file.

KEEP/N is the same, but does not append CR/LF to the output lines.

> KEEP ⊞   Starts a stretchable KEEP rectangle in the center of the screen. Use the arrow keys to move one corner of the highlighted rectangle to include the data you want to keep. Press ⊞ to nail that end down. Move the other corner to cover the desired KEEP area. Press ⏎ to display the selection coordinates on the command line.

> > *c1 c2 r1 r2*   Optional screen coordinates used to
> > specify a part of the screen, either
> > absolute or cursor-relative (as in BOX).

KEYS

Shows all currently defined keys. The first section is system
defined (hard coded) keys. The second is key definitions either
contained in the KEYS file or downloaded from the host.

Use PRINT to print the display.

KILL

Terminates the last program started by Anzio with a LAUNCH,
LAUNCH/S, RUN, or RUN/S command.

```
KRECEIVE AUTO-OFF
KRECEIVE AUTO-ON
KRECEIVE [options] [filespec]]
KRECEIVE/S [options] [filespec]]
```

Receives one or more files from a Kermit host system, or
controls Kermit auto-reception.

When Kermit auto-reception is on, a program running on the host
system can send (download) a file to the PC without the user
doing anything. This capability can be turned on or off with

> KRECEIVE AUTO-ON

and

> KRECEIVE AUTO-OFF

A file transfer is specified as:

> KRECEIVE *filename* ⏎

or

> KRECEIVE *wildcard* ⏎

If *options* are specified, they override the global options set in
Transfer/Transfer Options for this transfer session only. Options
start with "-" or "/".

➢ Options for Kermit, Xmodem, Ymodem, and Zmodem are shown
here. Some options are not applicable to some transfer protocols.

Possible options are:

-A	ASCII mode transfer
-B	Binary mode transfer
-C	XmodemCRC & Kermit Checksum
-D	Debug
-G	Xmodem1KG & YmodemG
-K	Xmodem1K & Kermit CRCk
-O	Zmodem WriteClobber
-U	Uppercase
-W *xxxx*	Windowing and window size
-1	Kermit CRC
-2	Kermit Checksum2
-3	Kermit CRC32
-8	Zmodem 8K
-V	Debug

If "/S" is included in the command, a result code is sent to the host system (or macro). The result code is always 2 characters followed by a Return. Possible result codes are:

00	Success
01	at least 1 file transferred, but at least one was not (not implemented yet)
02	file not found
03	file already exists
04	file skipped
05	cancelled by user
06	cancelled by host (not implemented yet)
07	transfer failed (started ok, but then failed for some reason)
08	protocol error
09	protocol timeout (not implemented yet)
10	critical unknown (not implemented yet)

See also KCOMMAND.

```
KSEND [options] filespec
KSEND/S [options] filespec
```

>   Sends one or more files to a Kermit host system. The *filespec*
>   can be either a single filename or a wildcard.

>   If *options* are specified, they override the global options set in
>   Transfer/Transfer Options for this transfer session only. Options
>   start with "-" or "/". For a description of available options, see
>   KRECEIVE, above.

>   If "/S" is included in the command, a result code is sent to the
>   host system (or macro). The result code is always 2 characters
>   followed by a Return. For possible result codes see KRECEIVE,
>   above.

```
LAUNCH [options] program [parameters]
LAUNCH/S [options] program [parameters]
```

>   Starts another Windows program to run alongside Anzio. Similar
>   to RUN, but Anzio does not wait for the other program to
>   complete. See also WINSTART.

>   Possible *options* are:

> | | |
> |-------|-----------------------------------|
> | /MIN | Minimize the program's window |
> | /MAX | Maximize the program's window |
> | /HIDE | Hide the program's window |
> | /*nnn* | The numeric value *nnn* is |

>   equivalent to that used in VBScript, etc.

>   Both *program* and *parameters* can contain environment
>   variables, coded as

>   ```
>   ${variable}
>   ```

>   Anzio will replace each environment variable with its contents.

>   See section 9.3 *Variable Substitution*.

>   If LAUNCH/S is used, Anzio will send a status return code back
>   to the host, indicating whether the *program* started successfully.
>   See RUN/S.

```
LINE [DELAY] n
```

> Rarely used. Sets the line turnaround delay. After an "unlock" is received from the host, Anzio delays its next transmission for the specified amount of time. *n* is a number from 0 to 65535 in units of 10 microseconds.

> The "unlock" protocol depends on the host operating system.

> If you are experiencing lockups, or problems with file transfer, particularly with NCR I-systems, you may need to experiment with LINE DELAY and DELAY settings.

> See also DELAY.

```
LOCK [ON] | OFF | TOGGLE
```
F4

> LOCK ON is used for the normal NCR I-system protocol, which 'locks' the keyboard when the host is not expecting entry. To temporarily unlock the keyboard, press F4 .

> LOCK OFF sends all characters as they are entered, never locking the keyboard. LOCK OFF is required by some programs, such as free-standing utilities and modems.

> **(UNIX)** Use LOCK OFF with UNIX systems.

```
LOG [drive:]directory
```

> Sets the default ("logged") disk drive and directory. The logged disk and directory are used for local file operations (DIR, OPENI, etc.) when no drive or directory is specified.

> The optional disk unit must be followed by a colon.

> Synonymous with the CD command.

```
LOGIN-WIZARD [ON] | OFF | TOGGLE
```

> Turns on or off the Login Wizard. When Login Wizard is ON, and a USERNAME and PASSWORD are set, Anzio will attempt to log you in as soon as the connection is made.

```
MAILTO
MAILTO variable=value [variable=value...]
```

> Causes Anzio to send an email, either automatically or interactively. In either case, an EMAILSETUP must have been done at least once in order to specify static parts of the email environment, such as the name of the email server.

MAILTO by itself will pop up a dialog box prompting the user for the parameters of the email message to be sent.

Or, specify various elements of the email with multiple *variable*=*value* pairs. Possible *variables* are:

To	Sets the recipient's email address.
Name	Sets the recipient's name (not possible in some email methods).
CC	Sets additional recipients.
BCC	Sets "blind carbon copies".
Subject	Sets the subject line
From	Sets the name of the sender
Fromaddress	Sets the email address of the sender
Notes	Sets the contents of the body text.
Attach	Sets one or more files to be attached
Show	ON or TRUE will prompt the user, OFF or FALSE will not.
Confirm	Same as "Show".

MAPISENDFAX *phone filename* [*profile*]

Uses Windows' messaging system (MAPI) to send a fax, to the phone number contained in *phone*. Windows decides how to "print" *filename* to the fax driver based on its file extension.

If a *profile* is included, and it matches a profile configured in MAPI, then MAPI will not prompt the user for any details, but will send the fax directly.

MCISENDSTRING *command*
MCISENDSTRING/S *command*

These commands provide an interface to Windows' Multimedia Control Interface, via the Windows command "mciSendString". The MCISENDSTRING Anzio command gives NO feedback on success; on an error, it will trigger an ONERROR macro if one is defined, or display "Function Error". The MCISENDSTRING/S version will send either "OK" or an error message back to the host.

Possible values of *command* are (somewhat) documented by Microsoft. Here is an example that records sound to a file. To initiate recording, do

```
MCISENDSTRING open new type waveaudio alias
    capture⏎
MCISENDSTRING set capture time format ms
    bitspersample 16 channels 2 samplespersec
    44100 format tag pcm⏎
MCISENDSTRING record capture⏎
```

To terminate recording and save the sound to a file, do:

```
MCISENDSTRING stop capture⏎
MCISENDSTRING save capture test.wav⏎
MCISENDSTRING close capture⏎
```

MENUBAR *c1 c2 r1 r2 c3 c4 off len* [*cols*]

> Superimposes a sliding menu bar system over the data presently on the screen, in one or more rectangles: (*c1*, *r1*) to (*c2*, *r2*) and (*c3*, *r1*) to (*c4*, *r2*). If a single-column menu is needed, just make *c3* equal *r1* and *c4* equal c*2*. If more than two columns are needed, use the *cols* parameters.

> When the user has selected an entry, Anzio returns the contents of the screen at offset *off* and length *len* of the item selected.

> See *MENUBAR Example*, page 56.

MENULANGUAGE *xxx*

> Sets the language that Anzio uses for its dialog boxes, menus, and error messages. Support for languages is provided via files (in the same directory as Anzio) named ANZIO*zzz*.DLL, where "*zzz*" is an abbreviation of the language, such as ANZIOESP.DLL for Spanish. These files are provided by Rasmussen Software and partner companies, and can not generally be created by users.

> The *xxx* language identifier can be a) the native name of the language, such as "español"; b) the name in the language of the Windows system, such as "Spanish"; or c) the Windows standard number for the language, such as "10" for Spanish. Entering zero, or encountering any error, will switch Anzio back to its built-in English mode.

`MERGE` *filename*

> Merges the key macros in *filename* with the macros in memory. Like READ, but does not clear existing macros, nor change the current key file name. If you SAVE after the merge, the merged set of macros is written to the current key file.

> Macros in *filename* will overwrite macros in memory only if they have the same name.

`MESSAGE` *string*

> Pops up a message box containing *string* and an OK button – useful in a defined key or a host command.

`MKDIR` *dirname*
`MKDIR/S` *dirname*

> Creates a new directory.

> MKDIR/S also sends a result code to the host:

> > 00   Completed successfully
> > 01   Error occurred

`MONITOR [ON] | OFF | TOGGLE`

> Sets monitor (diagnostic) mode. MONITOR ON displays all characters received from the host, including screen control codes. Non-printing characters are displayed as ASCII hex values.

> MONITOR OFF returns to normal Anzio terminal emulation.

> See also INTERPRET.

`NICE-MARGINS [ON] | OFF | TOGGLE`

> Turns on/off Print Wizard's "nice margins" option. If off, printing will start as high and as far to the left as possible. If on, some margins will be set depending on the number of rows and columns to be printed – see the Print Wizard manual.

```
OLE INIT program.title
OLE GET program.property
OLE program.property=value
OLE program.method [args]
OLE/S program.method [args]
OLE TERM
```

These commands provide an interface into the WINDOWS OLE system for calling and manipulating other programs. In Anzio, only one OLE object can be in use at a time. Note also that property names and method names must include the *program* part of the object name.

Begin an OLE session with

```
OLE INIT program.title
```

To get a property value, and return it to the host, do

```
OLE GET program.property
```

To set a property, do

```
OLE program.property=value
```

To call a method when you are not expecting a return value, do

```
OLE program.method [args]
```

where *args* are any arguments expected by the method.

To call a method that returns a value, and send that value to the host, do

```
OLE/S program.method [args]
```

where *args* are any arguments expected by the method.

To terminate the session, do

```
OLE TERM
```

```
ONERROR macro
ONERROR
```

This command tells Anzio how to handle errors in macro processing. Note that these are operational errors, not syntax errors. When an error occurs, the current macro (and any macros nested with the CALL command) is terminated, and *macro* is executed. For instance, if a macro:

F6 onerror E|F6 WAITFOR Login 5|guest|

is running, and the WAITFOR times out, the macro will be terminated, and the "E" macro will be run. Setting the ONERROR macro lasts only for the current program run; it is NOT retained in the parameters file. To return to normal default error handling, do just

```
ONERROR
```

ONSTARTUP *macro*
ONSTARTUP

Specifies a *macro* to be run on startup. The *macro* specified can be an alphanumeric name, or the keyname of the key such as "F12". This information is stored in the settings file, and will take effect the next time Anzio is run. To clear the setting, so NO macro will be run, do:

```
ONSTARTUP
```

A startup macro can also be specified in the command line, and that will take precedence over the ONSTARTUP macro, even if it doesn't exist.

If a startup macro is specified (by either method), the Login Wizard will not function. Your macro must then do whatever logging in is necessary.

OPENE *filename*
OPENE/N *filename*
OPENE/S *filename*

Opens a PC file for output, like OPENO (and variations) below, except in EXTEND mode. That is, new data will be added to the end of the existing data in the file.

Data can be written to the current output file in a number of ways, such as with KEEP, WRITE, CAPTURE, and PICK. Also, if an output file is open, passthrough print data received from the server will be written to that file instead of to the printer.

If *filename* does not already exist, and OPENE was used, that constitutes an error; an ONERROR macro will be triggered if one has been specified, otherwise an error message is presented.

If *filename* does not already exist, and OPENE/N was used, a new file will be created.

OPENE/S will send the following result codes to the host:

00	Completed successfully
01	File not found
02	Output file already open
03	File location is not in DOWNLOAD-
	LOCK directory, or other file error

See OPENO for more information.

```
OPENI filename
OPENI/S filename
```

Opens a PC file for input. If *filename* does not include a drive or directory name, the default drive and directory names are used (see LOG).

Only one input (and one output) file can be open at a time.

The file name mentioned can be a serial port, such as "COM2". Note that in some versions of Windows, if *any* printer driver is assigned to this serial port, it will interfere with Anzio accessing the serial port in this way.

The input file is typically used with TRANSMIT and READLINE.

OPENI/S also sends a result code to the host:

00	Completed successfully
01	File not found
02	Input file already open

```
OPENO filename
OPENO/N filename
OPENO/S filename
```

Opens a PC file for output from the host. If *filename* does not include a drive or directory name, the DOWNLOAD-DIR directory will be used if defined, otherwise the default drive and directory names are used (see LOG).

Only one output (and one input) file can be open at a time.

The output file is used for:

- CAPTURE – see *Data Capture*, page 84
- KEEP
- PICK
- WRITE
- File transfer – see *File Transfer*, page 86

- Passthrough printing – see *Passthrough Printing*, page 71

filename may be a DOS device name, such as COM2, or LPT2. If *filename* is LST:, output is sent to the printer set with the PRINTER command. Use OPENO LST: for file transfer from a host computer directly to the PC's printer.

A message is displayed if the file exists. Use DELETE to delete that file, or OPENO/N.

OPENO/N overwrites the file if it exists, without displaying a message.

OPENO/S sends a result code to the host, without displaying a message on the PC:

00	Operation successful
01	File already exists
02	An output file is already open
03	File or subdirectory error (including violation of DOWNLOAD-LOCK)

OVERLAY [*filename*]

Tells the Print Wizard feature to print with the specified overlay file. This might be a BMP, GIF, JPG, JPEG, TIF, TIFF, PCL, EMF, WMF, or PWML file.

If no *filename* is given, the overlay feature will be disabled.

PALETTE BG | BLBRIGHT | BLDIM | ULBRIGHT | ULDIM | FULL

Sets the palette. At any one time, Anzio can use 8-bit attributes or "full" (10-bit) attributes. With full attributes, Anzio can control 16 foreground colors, 16 background colors, blink, and underline. With 8-bit attributes, Anzio can display characters in a palette of 256 color/attribute combinations. For instance, it is possible to display blinking characters, or underlined characters, but not at the same time. This is often necessary for compatibility with earlier hardware designs. This command specifies which palette is in effect:

BG	Bright and dim backgrounds
BLBRIGHT	Blinking, with bright backgrounds
BLDIM	Blinking, with dim backgrounds

ULBRIGHT	Underlines, with bright backgrounds	
ULDIM	Underlines, with dim backgrounds	
FULL	Full palette of 16 foreground colors, 16 background colors, blink, and underline.	

PARITY EVEN | ODD | MARK | SPACE | OFF

> Sets the parity for communication over a serial connection.

PASSTHRU

> Puts Anzio into passthrough print mode, just as though it had received the appropriate escape sequence from the host.

PASSTHRU-CP *name*
PASSTHRU-CP/S *name*

> Sets the encoding scheme used for passthrough printing. Options for *name* are the same as for the CHARSET command, plus the following option:

>> NONE   Results in ISO or OEM depending on the terminal type.

> This setting has an effect only when printing with Print Level set to High, or when using Print Wizard.

> PASSTHRU/S sends a result code indicating whether it succeeded: 00 for success, 01 for failure. A failure could indicate that a particular codepage was not installed on this PC, for instance.

> See also CHARSET.

PASSWORD *text*
PASSWORD/S

> Sets or sends the host password. *text* is stored internally in an encoded format. Use a pipe character ("|") to represent a password of ⏎.

> The stored password is used by the Login Wizard, and is sent to the host with PASSWORD/S.

> PASSWORD/S sends the password to the host. The password was either set by the PASSWORD command, or as part of a command line parameter containing a URL.

See also USERNAME.

PASTE

Pastes text data on the Windows clipboard to the host. The CLIP command clips a region of screen text and puts it on the clipboard.

PDFSETUP

This will prompt the user for the options to be used when "printing" to a PDF.

PICK *c1 c2 r1 r2 type* [...]

Picks data off the screen and puts in the output file (see OPENO) for use by PC programs that understand comma-separated-values format. After the host displays a table to the screen, PICK can capture one or more columns of numbers or labels off the screen. The picked data is written to the current output file in CSV format.

Offsets can be cursor-relative – see BOX.

PITCH *n* | –1

Sets the type of sound that the PC will make when it receives a beep signal (control-G, hex 07) from the host.

> ➢ If your PC has a sound card but does not have speakers connected, Windows does not know that. Your only option is to force the beep to come from the PC motherboard's speaker.

PITCH –1 causes the beep to come from the PC motherboard speaker, rather than an installed sound card.

A value of *n* higher than 99 tells Anzio to try beeping at the indicated pitch. This will NOT work in Windows 95/98/ME.

Values between 0 and 64 cause Anzio and Windows to generate certain sounds, identified by name, as configured in the "Sounds" control panel applet. These values are:

0	Default sound
16	Critical stop
32	Question
48	Exclamation
64	Asterisk

```
PLAYMEDIA
PLAYMEDIA filename
PLAYMEDIA filename [CELL] x1 y1 x2 y2 [options]
```

Plays a multimedia file, such as an MP3, WAV, MID, or AVI file named *filename*, which can be a local file or a URL.

Only one media file can be played at a time. If a PLAYMEDIA is executed while a previous file is playing, the first playback is terminated. To stop the playback, use the command without a filename:

```
PLAYMEDIA
```

If the media file is video, such as an AVI file, you can optionally specify a location in the Anzio window for it to display, with

```
PLAYMEDIA filename [CELL] x1 y1 x2 y2
```

If CELL is given, then coordinates are in character rows and columns, starting at 1. If CELL is not given, coordinates are in pixels, relative to the top left corner of the top left character. For *x1* and *y1*, a value of -1 means the left or top edge. For *x2* and *y2*, a value of 0 means the right or bottom edge. If all four parameters are greater than zero, the video will be stretched, while maintaining aspect ratio.

If any text follows the *y2* above (that is, *options* are specified), that text will be passed on to the Windows mciSendString command. Examples are "repeat", "at time *xxx*", "from position *yyy*", etc. See Windows documentation of mciSendString for details.

This command is aborted by Grand Abort (Alt-A).

```
PLAYSOUND filename
```

Plays a WAV sound file named *filename*, which can be a local file or a URL. If the *filename* has an extension other than ".WAV", this command will be treated as PLAYMEDIA.

```
PORT n
```

Specifies the hardware communication port number – see *Configuring the Serial Port*, page 13.

```
PREVIEW [ON] | OFF | TOGGLE
```

Turns on or off Print Preview, which is part of the Print Wizard feature.

```
PRINT    [c1 c2 r1 r2]
PRINT/N [c1 c2 r1 r2]
```

> Prints the lines visible on the screen, ignoring video attributes. PRINT/N prints each line without adding carriage return-linefeed characters.

> The optional column and row offsets define the screen area to be printed, from column *c1*, row *r1* to column *c2*, row *r2*. Offsets can be cursor-relative – see BOX.

> If columns and rows are not given, Anzio will print the entire screen, unless the screen has a section highlighted, in which case it will print only the highlighted text.

> To print the screen during a REVIEW, enter P.

> To print displays generated by DIR, HELP, INTERPRET, KEYS, or TYPE, enter PRINT ⏎.

```
PRINTDOCNAME name
```

> Sets the document name of the next passthrough-print job to be received. This can help to determine the name of a generated PDF, for instance.

```
PRINTER name
PRINTER WPRN
```

> Identifies the printer or output driver used for the PRINT and PRINTER-SETUP commands, and for passthrough print.

> PRINTER WPRN tells Anzio to use its Windows printer driver.

> If *name* is a file, that file is opened for output. If the file exists, it is overwritten.

> If *name* contains the string "://", it is treated as a SUPERFILE designation, such as "PDF://" to specify creation of an Adobe PDF file.

```
PRINTER-SETUP
```

> Causes Anzio to bring up its standard Printer Setup dialog box, which lets you choose how printing jobs should be handled.

```
PRINTER-SETUP text
```

> Sends the characters in *text* directly to the printer.

text may also include control characters. To embed a control character in *text*, preface it with CTRL P . For example, to send ESC Ctrl-N to your printer:

PRINTER-SETUP CTRL P ESC CTRL P CTRL N ⏎

As another example, suppose you want to define the F12 key to print the screen, and then eject a page. Use the PRINT command to send the screen to the printer, and then send the printer a form-feed character (Ctrl-L):

DEFINE F12 F6 PRINT |
    F6 PRINTER-SETUP CTRL L | ⏎

Note that a vertical bar | ends each command.

To embed any hex character in *text*, enter CTRL P followed by hex 10 and the two-character ASCII hex value of the character. The hex 10 represents a second Ctrl-P, used internally for quoting ASCII characters. For example, to embed a Return character (hex 0D), enter:

CTRL P 100D

PRINTFILE [*/charset*] *filename*

Copies *filename* to the current PRINTER. *filename* can be a local file or a URL, causing Anzio to fetch the file from a web or FTP server.

The file is assumed to be in the OEM ("DOS") character set. This can be overridden with a *charset* value, following a slash, as follows:

/I	ISO character set
/U	Unicode UTF-8
/O	OEM

If you are using the Windows printer driver (PRINTER WPRN) and Anzio's Print Wizard is enabled, the Print Wizard examines the data in *filename* and sets the line spacing, character size, and margins to make the document fit on the page.

If the *filename* has an extension indicating it is a bitmap (.BMP, .JPG, .JPEG, .TIF, .TIFF, or .GIF), or a metafile (.WMF, .EMF, .SPL), the file will be printed as a full page, as large as possible on the page.

PRINTFONT [*name*] [*size*] [BOLD]

> Used with the Windows printer driver (PRINTER WPRN) to set the printer font name and size. Enclose *name* in quotes if it contains one or more spaces. *size* is either *height* or *height*x*width* (specified in points). Specifying BOLD selects a bold font.

PRINTFONT-ALT *fontnames*

> Tells Anzio what alternate fonts should be used for the printer, to print characters not found in the primary font. This setting applies only when printing with Print Wizard. The *fontnames* can be a list of fonts by their Windows names, separated by commas. Fonts will be tested in the order indicated. To tell Anzio NOT to use a particular font, put a minus sign in front of its name. The last entry should generally be "auto", telling Anzio to look through any other fonts on the Windows system. The default is simply "auto".

PRINTINIT *filename*

> Specifies a print-initialization file. When Anzio receives a passthrough print job, if a file with this name exists, it is read and inserted at the beginning of the print job. The file can be built with a text editor, such as Notepad, and can contain text to print, escape sequences for printer control, or Print Wizard Markup Language. If the named file does not exist, no error will occur, but it will be noted in the Debug Printing window, if enabled.

PRINTLEVEL WIZARD | HIGH | LOW | SPOOLER | RAW

> Used with the Windows printer driver (PRINTER WPRN) to set the print level.
>
> PRINTLEVEL WIZARD turns on the Print Wizard
>
> PRINTLEVEL HIGH indicates high-level print.
> PRINTLEVEL LOW indicates low-level print.
> PRINTLEVEL SPOOLER indicates spooler-level print.
> PRINTLEVEL RAW indicates raw-level print.

PRINTLINE [*text*]
PRINTLN [*text*]

> Writes *text*, if specified, to the printer. Then writes a return and a linefeed to the printer.

```
PRINTLOW [ON] | OFF | RAW | SPOOLER
```

> Used with the Windows printer driver (PRINTER WPRN) to set the print level. See also PRINTLEVEL.
>
> PRINTLOW ON indicates low-level print.
> PRINTLOW OFF indicates high-level print.
> PRINTLOW SPOOLER indicates spooler-level print
> PRINTLOW RAW indicates raw-level print.

```
PRINTMACRO [macro]
```

> Specifies a code-activated-switch macro. The stream of print data coming from the host in passthrough print can contain control characters control-A (hex 01) through control-F (hex 06), which can trigger Anzio to initiate one of six macros starting at *macro*. For instance, if you do
>
> > ```
> > PRINTMACRO a
> > ```
>
> then when the passthrough print data contains a control-A, Anzio will CALL the "a" macro; when it contains a control-B, Anzio will call the "b" macro, etc., up to control-F. When Anzio calls the macro, it exits from the passthrough print routine.
>
> Typically, these macros will each switch to a different printer, and then use the PASSTHRU command to return to doing passthrough print. This way, control codes embedded in the data stream can effectively switch from one printer to another.

```
PRINTTOFILE [ON] | OFF | filename
```

> Tells Anzio whether to print to file. This is equivalent to checking the "Print to file" box in the Printer Setup dialog. However, with this command you also have the option of specifying the *filename* to be printed to, so the operator will not be prompted for that.
>
> Note that this file will contain printer control codes as included by the Windows printer driver.

PROMPT-ID

> Displays a dialog box to prompt the user for login and password,
> with no default values. Values entered are stored in Anzio's
> internal username and password variables, and are stored in the
> settings file if you save settings. Values can be used by the
> Login Wizard at the time of the next connection, or they can be
> used in macro-programmed logins, using the commands
> USERNAME/S and PASSWORD/S.

> If the user clicks Cancel, and an ONERROR macro is defined, it
> will be started.

PROMPT/S [*caption* [*text*]]

> Prompts the user for a string of alphanumeric input, using a
> Windows dialog box. The result is sent to the host (followed by a
> <return>), or nested into another command (without a <return>).

> If a *caption* is given, it will appear in the top line of the dialog box.
> If *text* is also given, it will appear above the user input prompt.

PURGE

> Clears Anzio's "received file" buffer. This buffer gets data from
> either CAPTURE or a file transfer.

> Data in this buffer is waiting to be stored in an output file. If there
> is no output file (see OPENO), the buffer will contain a backlog of
> data. The HELP screen displays the buffer status.

RAW-SETUP [DEFAULT]

> This command allows you to specify a particular port (parallel or
> serial) to be used whenever Anzio prints at the RAW level, and
> to configure that port.

> RAW-SETUP first prompts for the name of the port. If this field is
> empty, Anzio will derive the port name from what is shown under
> File/Printer Setup. Or, you can enter a port name such as "LPT2"
> or "COM1".

> Then, if the port as specified or derived is a comm port, Anzio
> will display a dialog box allowing you to specify baud rate, parity,
> etc.

> To tell Anzio to use the printer specified in Printer Setup for Raw
> printing also, do

> > Raw-setup default

READ *filename*

>   Reads in a file of defined keys and macros, replacing those in
>   memory. *Filename* becomes the new default key file name used
>   by SAVE.

>   Note that *filename* can be a URL, meaning Anzio will read its key
>   definitions from an HTTP[S] or FTP server.

>   This *filename* gets stored in the settings file (if you save your
>   settings), and will be used to specify where the key definitions
>   will be loaded from every time you run Anzio.

>   See also MERGE.

READLINE/S *endmacro*

>   Reads one line of the currently open input file (see OPENI), and
>   sends its output to the server, or to a nesting command. The
>   contents of the input file must be line oriented, and are assumed
>   to be in the ISO character set.

>   When the end of the input file is reached, if *endmacro* is
>   specified, then that macro will be triggered. Otherwise, an error
>   condition will exist, which will trigger an ONERROR macro if one
>   is in place.

>   When combined with the REPLACE/S command, this allows a
>   file to be read a line at a time, manipulated, and part or all of it
>   sent to the server.

RECEIVE CODED

>   Used when transmitting coded files between PCs.

RECEIVE QUIET [ON] | OFF | TOGGLE

>   Controls whether file transfer data and host commands are
>   displayed.

>   RECEIVE QUIET ON does not display transfer data, and so is
>   slightly faster for file transfer. This setting also affects functions
>   sent from the host.

>   RECEIVE QUIET OFF is the default, and all transferred data is
>   displayed on the screen.

```
RECONNECT [ON] | OFF | STAY
```

> Specifies reconnection behavior when the host system drops its end of the connection, for example, after you log off from a UNIX host. Does not apply to direct serial connections, but does apply to modem (TAPI) connections.
>
> RECONNECT ON causes Anzio to try a new connection, leading to a new login prompt.
>
> RECONNECT OFF causes Anzio to exit when the connection ends.
>
> RECONNECT STAY causes Anzio to keep running, but in a disconnected state.

```
RECORD [ON | OFF | TOGGLE]
RECORD CANCEL
RECORD keyname
```

> These commands control Anzio's macro recording capability. To begin recording a macro, do simply RECORD. Anzio will start recording and saving your keystrokes. To cancel recording, and discard saved keystrokes, do RECORD CANCEL.
>
> If you do RECORD OFF, Anzio will stop recording. It will prompt you for the key you wish to assign the saved keystrokes to.
>
> If you do RECORD *keyname*, Anzio will stop recording and assign the macro to the key identified by *keyname*.

```
REGSSH [RESET]
```

> Anzio can register itself to Windows as the official SSH client. Then, whenever the user launches a URL that starts with ssh://, such as from a web page, Windows will start Anzio.
>
> REGSSH sets this version of Anzio as the official SSH client.
>
> REGSSH RESET sets Windows back to its default state (which is having NO SSH client).

```
REGTELNET [RESET]
```

> Anzio can register itself to Windows as the official telnet client. Then, whenever the user launches a URL that starts with telnet://, such as from a web page, Windows will start Anzio. This is also known as "browser integration".
>
> REGTELNET sets this version of Anzio as the official telnet.

REGTELNET RESET sets Windows back to its default state.

RENAME *oldname newname*

Renames a PC file from *oldname* to *newname*.

REPLACE/S *search-pattern replace-pattern content*

Does a "regular expression" replacement on *content*, and returns the result to the host or to a nesting command. Contact us for more information on syntax.

RESET

Resets some terminal emulation parameters: character sets, attributes, and wrap mode.

RESTART [/NOSAVE]

Restarts Anzio; that is, it starts a new session of Anzio and terminates this current session. If /NOSAVE is specified, the restart is immediate, with no saving of settings and no user interaction.

If /NOSAVE is NOT specified, and settings have changed, the user may be prompted to save settings, depending on Anzio's configuration, the same as when quitting.

RESTOREPARAM [*section*]
RESTOREPARAM [*section*]*variable*

➢ In this command the square brackets are part of the command, and are not used to indicate optional content.

Restore an entire *section*, or just one *variable*, to the value(s) saved in a corresponding SAVEPARAM.

RETRANSMIT

Resends the last function response code, or, during file transfer, resends the last file record transmitted.

REVIEW

Allows you to see old screen data that has scrolled off the top of the screen. See *Using Review Mode to See Previous Data*, page 40.

RRECV [*options*] [*hostname* [*pcname*]]

Receives (downloads) a file using the RTunPlus protocol.

Options can be "ASCII" or "-A" for an ASCII (text mode) transfer, or "-B" for binary (the default).

If a *hostname* is not given, you will be prompted for it.

```
RSEND [options] [hostname [pcname]]
```

Sends (uploads) a file using the RTunPlus protocol.

Options can be "ASCII" or "-A" for an ASCII (text mode) transfer, or "-B" for binary (the default).

If a *hostname* is not given, you will be prompted for it.

```
RUN   [options] [program [parameters]]
RUN/N [options] [program [parameters]]
RUN/S [options] [program [parameters]]
```

Runs another Windows or DOS *program*. When *program* ends, control returns to Anzio, with the screen intact. For example, you can change from terminal emulation to word processing and then back.

Before restoring the screen, Anzio prompts you to press any key. This is so you are able to see *program*'s output if necessary.

Anzio starts *program* and then goes into an inactive state until *program* finishes, for compatibility with DOS versions of Anzio. Use LAUNCH or WINSTART to run a concurrent program with Anzio.

RUN/N does not require a keystroke to restart Anzio after *program* ends.

RUN/S also sends *program*'s exit code to the host.

Possible *options* are:

/MIN	Minimize the program's window
/MAX	Maximize the program's window
/HIDE	Hide the program's window
/*nnn*	The numeric value *nnn* is equivalent to that used in VBScript, etc.

If you do not specify *program*, the RUN commands call the DOS command interpreter (COMMAND.COM or CMD.EXE). You can then enter any DOS commands. Use EXIT to exit DOS and return to Anzio.

Some programs must be started in their home directory. Use LOG to change to the required directory before using the RUN command.

program can be a Windows or DOS command or an application. For applications, enter the application name including the ".exe" or ".COM" suffix.

If *program* does not include a suffix, Anzio calls COMMAND.COM or CMD.EXE to find *program*, somewhere in the PATH. For example:

```
RUN COPY MYFILE PRN⏎
```

To allow *program* to access the results of a recent file transfer, the output file must first be closed (see CLOSEO). The HELP screen will show you if you have an output file open.

Both *program* and *parameters* can contain environment variables, coded as

```
${variable}
```

See section 9.3 *Variable Substitution*.

RZ

An alias for ZRECEIVE. See ZRECEIVE.

SAVE [*filename*]

Saves all currently defined keys (macros) to a PC file, overwriting the file's previous contents. If *filename* is not specified, Anzio will use the file which was read at startup, if any.

SAVEPARAM [*section*]
SAVEPARAM [*section*]*variable*

➢ In this command the square brackets are part of the command, and are not used to indicate optional content.

Saves Anzio's settings for an entire *section* or a single *variable* within a section, in memory, so that it can later be restored with RESTOREPARAM. Storage is only one level deep.

Section and variable names are as stored in Anzio's settings file.

SCREENMODE *option*

Sets the screen driver. SCREENMODE/S, below, reports screen mode parameters to the host.

option is one of:

132	132 columns wide
80	80 columns wide
*C*x*R*	Sets the window to the indicated size, for example, 132x25. In a network connection, if the new size is supportable, sends the new size to the host.

SCREENMODE/S

Reports several screen parameters to the host, formatted as one line:

TMMLLLCCCWWW*screenmode*

T (type)	C for color, M for mono, B for B&W, blank if unknown
MM (mode)	BIOS hex mode
LLL (lines)	Number of lines on screen
CCC (cols)	Number of physical columns
WWW (width)	
	Number of logical columns
screenmode	
	The SCREENMODE as displayed on the HELP screen

See also SCREENMODE above.

SCREENSIZE/S *param*

This causes Anzio to respond to the host with a numeric value containing information on the screen, according to *param* (all units are in pixels):

CLIENT.X	sends width of client (interior of window)
CLIENT.Y	sends height of client
WINDOW.X	sends width of window (including borders)
WINDOW.Y	sends height of window (including borders, menu, etc.)
MAX.X	sends monitor width
MAX.Y	sends monitor height

```
SCRIPT
SCRIPT filename [E:lang][M:module][T:timeout][args]
```

>   Runs an external script, such as in VBScript or JavaScript. If the
>   command is simply
>
>   ```
>   SCRIPT
>   ```
>
>   then Anzio will provide an interactive approach to debugging
>   your script. Ordinarily, though, you will run a command with at
>   least
>
>   ```
>   SCRIPT filename
>   ```
>
>   This runs the module "Main" in the script file *filename*, in the
>   VBScript language, with no timeout and no arguments. Optional
>   parameters are:

E:*lang*	Specifies the language of the script. Options are "vbscript" and "javascript".
M:*module*	Specifies the module (subroutine) within the script to run.
T:*timeout*	Specifies a timeout value
Args	One or more arguments that are passed to the script module

>   If the script returns anything (a string), Anzio will process it as an
>   Anzio command.

```
SCROLL [ON] | OFF | TOGGLE
```

>   Controls how a host "clear screen" request is interpreted.
>
>   SCROLL ON ignores "clear screen", instead scrolling the current
>   lines up and off the live screen. Use REVIEW to see the
>   scrolled-up data.
>
>   SCROLL OFF clears the screen when requested, erasing all
>   data from the previous screen. REVIEW of this data is not
>   available.

```
SCROLL-LOCK [ON] | OFF | TOGGLE
```

>   Enables or disables the `SCROLL LOCK` key. The `SCROLL LOCK` key suspends
>   screen display of data from the host, useful when data is arriving
>   too rapidly to read. See also HOLD.

```
SEND
```

>   Sends an image of the screen to the host.

```
SEND c1 c2 r1 r2
```

> Sends a line of data from the screen to the host, from column *c1* to column *c2* on line *r1* through *r2*.

> Positions can be cursor-relative – see BOX.

```
SEND "string"
SEND/N "string"
```

> Sends the *string* to the host. SEND adds an ⏎, SEND/N does not.

> The quote marks are necessary to distinguish this command from other forms of SEND.

```
SENDFILE [options]
```

> Sends a file to the server. The protocol used will the default file transfer protocol, as configured in the menu system. The *options* are passed along to the appropriate "send" command. So for instance, if the default file transfer protocol is Zmodem, the command "SENDFILE *options*" would get translated to "ZSEND *options*", and that command would get executed.

```
SET variable=value
```

> Sets a user-defined string *variable* to the string *value*. Variable names can be any string not containing "=", and are case independent. The *value* can be any string.

> Variables set in this way can be dereferenced with "${*variable*}" in most commands. See section 9.3 *Variable Substitution*.

```
SETCOLOR n | NORMAL
```

> Sets the working color to color index *n*, where *n* is either a decimal number or a hex value represented as '*xx*H'. Does not affect the attribute/color table (see COLOR).

> Use SETCOLOR NORMAL if you accidentally set a strange color.

```
SETCOOKIE URL name value
```

> Sets an Internet cookie, such as a web browser might use. Sets the cookie associated with *URL* and a *name* to a particular *value*.

> Cookies can be used to communicate between Anzio and other programs on the same PC.

See also GETCOOKIE.

SETKEYBOARD *n*

>   Tells Windows to set the keyboard input language to the numeric
>   value *n* as per Windows definitions. English, for instance, is 409.

SETPARAM [*section*]*variable*=*content*

> ➢ In this command the square brackets are part of the command, and
>   are not used to indicate optional content.

>   Sets a single Anzio configuration setting for the setting *variable*
>   in *section*. Variables and sections are as used in Anzio's
>   settings file, which can be examined (and changed) in Notepad.

>   Note that this is a low-level command; setting a configuration
>   item this way might not change associated items in the same
>   way that doing another function in this reference section would
>   do.

SFTP

>   Initiates an *interactive* Secure File Transfer Protocol (SFTP)
>   session over an existing SSH connection. The dialog box
>   provides an easy point-and-click interface that allows you to
>   move files easily between the PC and the host system. Unlike
>   the FTP command, the SFTP command allows transfer only with
>   the same host you are already connected to.

SFTPDELETE *hostfile*
SFTPDELETE/S *hostfile*

>   Causes Anzio to use SFTP commands to delete a file on the
>   server you are currently connected to.

>   *hostfile* can be a name or a pathname (with either kind of slash).

>   The "/S" causes a status result to be sent to the host ("00"
>   means success).

```
SFTPGET [hostfile] [pcfile]
SFTPGET/S [hostfile] [pcfile]
SFTPPUT [hostfile] [pcfile]
SFTPPUT/S [hostfile] [pcfile]
```

These commands do *programmed* Secure File Transfer Protocol (SFTP) transfers over an existing SSH connection. They allow AnzioWin to work as an SFTP client, although commands can be issued from the host (through Anzio's usual command protocol).

The "/S" causes a status result to be sent to the host ("00" means success). Anzio's DOWNLOAD DIRECTORY will affect downloads (FTPGET[/S]).

You can enter an empty *hostfile* by using a pair of double quotes: "". Then the hostfile name will be assumed to be the name part (without the path) of the *pcfile*.

If neither *hostfile* nor *pcfile* is not specified, you will be prompted to select a PC filename, and the host file name will be assumed to be the same (without the path).

```
SFTPREGET hostfile [pcfile]
```

Similar to SFTPGET, this command will "re-get" a file via SFTP from the server. That is, it will download it only if necessary.

If the file already exists on the PC, Anzio will query the host to determine the date/time of the host file. If it is not newer than the PC file, no transfer will occur.

If the file does NOT exist on the PC, or the host file is newer, the file will be transferred.

```
SHOW-COMMAND [ ON ] | OFF | TOGGLE
```

If ON, commands from the host, commands in macros, and their responses sent back to the host will be shown on the screen. If OFF, all that will happen invisibly.

```
SIGNAL n
```

Sends a signal number *n* to the server, according to Unix definitions.

On a telnet connection, *n* can be 1, which is "SIGHUP" or "hangup".

On an SSH connection, *n* can be 1 (hangup), 2 (interrupt), 3 (quit), 9 (kill), or 15 (term). However, in the Open-SSH daemon, as of version 4.1, these signals are ignored.

SLEEP *hh mm ss*

Tells Anzio to sleep until the given PC clock time is reached. For example, you could set Anzio to wake up and dial a remote computer at a preset time.

Enter ⌜ALT⌟ ⌜A⌟ to interrupt a SLEEP and awaken Anzio.

SPEAK
SPEAK "*text*"
SPEAK *c1 c2 r1 r2*

These commands do text-to-speech, using Windows' speech interface (SAPI). SAPI is included in some versions of Windows, and is downloadable from Microsoft's web site. The configuration of SAPI, including the audio devices to be used, the voice characteristics, and the language, are set in the Windows control panel.

The command SPEAK will read the entire screen, or the highlight text if a highlight is defined. If the argument after SPEAK is in quotes, it is considered as literal *text*; that *text* will be spoken. If the arguments following the word SPEAK are not in quotes, they are assumed to be the coordinates of a rectangular area of the screen — that part of the screen will be read.

SPOOLER [*name*]

Tells Anzio what printer to use when printing with Print Level = Spooler. This can be a different printer from what is used with Print Level = High. Note that when Print Wizard is being used, and it detects that a print job contains escape codes (and it is not translating PCL codes), it will switch to Spooler mode, which will then use the printer identified here.

Use the Windows printer name for *name*. To tell Anzio that the SAME printer should be used for Spooler as for High, omit *name*.

SSHAUX-ADD *num* [*command*] *portstring*

Adds an SSH backchannel serial port.

`SSHPORT-ADD` *`localport hostport host`*

> Adds an SSH local port forward. Anzio will monitor the PC's TCPIP port *localport*. If a connection is made there, it will be forwarded via the SSH protocol to the port *hostport* on the machine whole name or IP address is indicated by *host*.

> To allow port forwarding in general, you must open the settings file in an editor (such as Notepad) and change the settings ssh-forward to have a value of "1".

`SSHPRINT-ADD` *`num command`*

> Adds an SSH backchannel print.

`STATUS [LINE] [ON] | OFF`

> Tells Anzio whether to put status information on the bottom line of the screen. Status includes CAPS lock, NUM lock, keyboard LOCK, and communication errors.

`STAYALIVE` *`nn`*

> Sets the stayalive interval to a certain number (*nn*) of seconds. When Anzio is connected via telnet or SSH, and STAYALIVE is non-zero, Anzio will send a "do-nothing" packet every *nn* seconds. This can keep a router, firewall, etc. from disconnecting the session during periods of inactivity.

`STOP`

> Stops the Anzio program and returns to the operating system. Same as END.

`STOP [BITS] 1 | 1.5 | 2`

> Sets the number of stop bits in the serial communication protocol. Usually 1.

`SWITCHTO /NEXT`
`SWITCHTO /PREV`
`SWITCHTO` *`title`*

> These commands tell Windows to switch "focus" to a different Windows program that is currently running, or another instance of Anzio. SWITCHTO /NEXT will switch to the "next" instance of Anzio (in order by process number). SWITCHTO /PREV will switch to the previous instance.

Any argument not starting with a slash is assumed to be the title of another Windows program, as shown in its title bar. If the title contains spaces, use quotes around the *title* argument.

```
SWITCHTO/N /NEXT
SWITCHTO/N /PREV
SWITCHTO/N title
```

Same as the equivalent SWITCHTO above, except that no message is given upon failure, such as if no other session of Anzio is running.

```
SZ
```

An alias for ZSEND. See ZSEND.

```
TAB i j k ...
```

Sets tab stops at columns *i*, *j*, *k*... Tab column numbers are entered in ascending order, separated by any non-numeric character.

Tab stops can also be set with control codes from the host computer.

When the host system sends a TAB or BACKTAB control character, Anzio moves the cursor forward (backward) to the next (previous) tab stop.

See also TAB CHARACTER and TAB ON.

```
TAB RESET
```

Resets all tab stops to every 8 columns, starting at column 1.

```
TAB CHARACTER x
```

Sets the character sent for the Tab key.

(I-systems)   With TAB ON, when you press the Tab key, the TAB CHARACTER (usually space) is sent enough times to move the cursor over to the next tab column.

```
TAB [ON] | OFF
```

TAB ON uses the TAB CHARACTER for the Tab key's code.

TAB OFF uses the standard ASCII TAB character for the Tab key.

```
TAKEFOCUS
```

> Causes Anzio to take focus back, such as after doing a
> WINSTART to start another program. Note that you may need to
> cause some kind of delay after WINSTART before doing
> TAKEFOCUS.

```
TERM termtype
```

> Sets the terminal type to be emulated:

N7900	NCR 7900 Model 1
N7901	NCR 7901
VIEWPOINT	ADDS Viewpoint (same as N7901)
VT100	DEC VT100
VT220	DEC VT220
VT320	DEC VT320
VT420	DEC VT420
WYSE50	Wyse 50
WYSE60	Wyse 60
T160E	Innovative Software's multi-language extension to VT220
C332	Versyss C332
SCOANSI	Console of SCO UNIX
TV965	Televideo 965
AT386	Console of various AT&T and UnixWare UNIX systems
ANZIO	Anzio's own definition, based on VT220
IBM3161	IBM 3161
ADDS200	ADDS Regent 200

> ➢ Be sure that Anzio's TERM setting agrees with the host computer's
> expected terminal type.

```
TERMNAME name
```

> Sets a TERM variable that will be sent to the host as part of the
> telnet, SSH, or rlogin protocol, during the initial connection. If no
> TERMNAME is specified, Anzio will send a name based on the
> TERM type.

With some host systems, it is possible to specify several choices for TERMNAME, separated by semicolons, with the most desirable setting first. For instance:

```
TERMNAME vt320;vt220;vt100
```

During the option negotiation that occurs at the beginning of a telnet session, Anzio and the host system will choose which option works best for both.

TIFSETUP

Prompts the user for the options to be used the next time a job is "printed" to a TIF file.

TIME/S [*format*] [*offset*]

Sends the PC clock time to the host, by default in the format HH:MM:SS⏎. If *format* is given, it is used as a character pattern to indicate the format which should be used. Possible characters are:

h	Displays the hour without a leading zero (0-23).
hh	Displays the hour with a leading zero (00-23).
m	Displays the minute without a leading zero (0-59).
mm	Displays the minute with a leading zero (00-59).
s	Displays the second without a leading zero (0-59).
ss	Displays the second with a leading zero (00-59).
z	Displays the millisecond without a leading zero (0-999).
zzz	Displays the millisecond with a leading zero (000-999).
am/pm	Insert an "am" or "pm".

If an *offset* is given, that number of seconds is added to (or subtracted if negative) the current time, and the resulting time is sent.

TIMEOUT *n*

Sets the file transfer timeout, where *n* is in tenths of a second.

If Anzio sits locked for the specified time, it will a) beep, b) unlock itself, and c) RETRANSMIT. This can get a file transfer going again after some types of communication errors.

TITLE *string*

Sets Anzio's window title.

TOOLBAR [ON] | OFF | TOGGLE

Turns on or off the toolbar at the top of the window.

TOOLBUTTON *num macro description*

Adds or changes the content of buttons on the toolbar.

Num is the number of the button, from 1 to 20.

Macro is the name of the macro string to be triggered by the tool button.

Description is the text to be displayed on the button.

TRACK-WINDOW [ON] | OFF | TOGGLE

Tells Anzio whether or not to save and restore its last window positions. This applies to the main window, the Print Preview window, and the FTP/SFTP window.

TRACK-WINDOW ON works between sessions only if you SAVE the current settings during a session or when you exit. But changing the size or position of one of these windows will also constitute a settings change, so Anzio will prompt you to save the settings upon exit (unless that feature is disabled).

TRANSFER-OPT

Brings up the Protocol Options dialog box for setting file transfer options.

TRANSMIT *filename* [*param*] [TRAILER *string*]
TRANSMIT [ON] | OFF

Controls file transmission according to the host protocol. See *File Transfer*, page 86.

TRANSMIT *filename* opens the file, transmits it to the host, and closes it. If *param* is not given, the file is transmitted one line at a time, with each line followed by a Return. Possible values of *param* are:

SINGLE	Sends one line, then suspends until an XN (transmit next) command happens.
CRC	Special mode for Rasmussen Software's UFT protocol.
LONG	Breaks the line into pieces with a length indicator.

TRANSMIT ON sends the last OPENI input file to the host, one line at a time.

TRANSMIT OFF ends transmission. TRANSMIT OFF is automatically called after sending the last line of the input file.

TRANSMIT TRAILER specifies an end-of-file *string* to be sent to the host after the last transmitted file record. For example, the host software may expect "END$" or "\".

To specify a DEL character as the trailer, enter *string* as either CTRL P DEL or ALT 127.

TYPE *filename*

Displays a file on the screen. Non-printing characters are shown as hex codes in reverse video.

To verify a received file after transmission, use CLOSEO first, then TYPE the file.

You can PRINT the displayed information.

UNICODE/S [*num*]

Provides a way to enter a character by its hexadecimal Unicode value. If *num* is given, it is assumed to be in hex, representing a character. If *num* is NOT given, Anzio will prompt the user for the value.

The character value is converted according to the current character set, and sent to the host.

UPPERCASE [ON] | OFF | TOGGLE

Like the CAPS LOCK key, if UPPERCASE is on, alphabetic keys are changed to upper case before being sent to the host.

UPPERCASE TOGGLE changes between UPPERCASE ON and UPPERCASE OFF.

USERNAME *name*
USERNAME/S

> Sets or sends Anzio's internal username, which is used by the Login Wizard and possibly the various FTP commands.

> A username can also be specified in Anzio's command line (see *Names – Settings File Name or Telnet Host Name*, page 192).

> USERNAME/S sends the current username to the host system.

VERSION
VERSION/S

> Displays (VERSION) or sends (VERSION/S) Anzio's version number.

WAIT *nnn*

> This command simply tells Anzio to wait a certain amount of time before sending anything else to the host. The *nnn* parameter is measured in tenths of a second.

WAITFOR *string* [*timeout*]
WF *string* [*timeout*]

> Causes Anzio to wait for the given *string* of characters to appear on the screen, or for *timeout* seconds to elapse. Enclose *string* in quotes if it contains a space. The default *timeout* is 86400, or 24 hours.

> WF is an abbreviation for WAITFOR.

> To interrupt a WAITFOR, press ALT A .

> If a WAITFOR times out, and an ONERROR macro is in effect, that macro will be INVOKEd.

WAITQUIET *seconds*

> This causes a macro to suspend operation until the host is "quiet" (not sending any data) for a certain number of *seconds*. This might be used, for instance, to have a macro CAPTURE data coming from the host. It could begin capturing, then do a WAITQUIET 3. When Anzio detected no host output for 3 seconds, the macro could continue and turn off CAPTURE.

```
WINDOW c1 c2 r1 r2 [FILL] [BOX | DOUBLE]
```

> Opens a character-mode "window" on the screen, from column *c1*, row *r1* to column *c2*, row *r2*. The original screen contents are saved in memory and later restored with the WINDOWCLOSE command.
>
> Positions can be cursor-relative – see BOX.
>
> The options are:

FILL	Fill the area with spaces.
BOX	Draw a single-line box around the new window.
DOUBLE	Draw a double-line box.

```
WINDOW option [option ...]
```

> Changes the Anzio window's position on the screen or other characteristics.
>
> The options are:

ZOOM	Maximizes the window.
MINIMIZE	The window will appear only as an icon on the task bar.
RESTORE	Takes the window out of zoomed or minimized state.
STYLE SUPERZOOM	Maximizes the window and removes the title bar and edges.
STYLE DEFAULT	Sets the window to its default state.
STYLE *nnn*	Sets the style to a numeric value, according to Windows standards.
MENU [ON] | OFF | TOGGLE	Turns on or off the main menu.
ROTATE [0 | 90 | 180 | 270]	Rotates the video display on PCs that support this feature, especially the TabletPC.
ROTATE TOGGLE	Switches display rotation between 0 and 270.

WINDOWCLOSE

> Following a WINDOW command, closes the window and restores the screen.

WINPRINT *filename* [*printer-name*]

> Asks Windows to print *filename* using the program associated with its file extension, in the same way as Windows Explorer. This uses the "printto" shell command.

> If *printer-name* is given (possibly surrounded by quotes), the file is printed on that printer; otherwise, the file is printed on the Windows default printer.

> *filename* can contain environment variables, coded as

>> ${*variable*}

> See section 9.3 *Variable Substitution*.

WINSTART *filename [params]*

> Asks Windows to start or open *filename*, which is either a program file or a data file. If *filename* is a data file, it is opened using the program associated with its file extension, in the same way as Windows Explorer.

> *filename* can contain environment variables, coded as

>> ${var}

> See section 9.3 *Variable Substitution*.

> If *filename* is a program, it can be started with command line parameters as in *params*.

WRITE *text*

> Writes *text* to the current output file, opened by OPENO.

> *Text* may include control characters. To embed a control character in *text*, preface it with CTRL P , for example CTRL P ESC .

> To embed any ASCII character in *text*, enter CTRL P followed by hex 10 and the two-character ASCII hex value of the character. The hex 10 represents a second CTRL P , used internally for quoting ASCII characters. For example, to embed a Return character (hex 0D), enter:

>> CTRL P 100D

```
WRITELINE [text]
WRITELN [text]
```

> If *text* is specified, it is written to the current output file. Then a return and linefeed are written to the current output file.

```
XN
```

> Transmits the next line of a file to the host. Used with TRANSMIT SINGLE.

```
XRECEIVE [options] [filespec]]
XRECEIVE/S [options] [filespec]]
```

> Receives one or more files from the host system, using the Xmodem protocol.

> A file transfer is specified as:

> > XRECEIVE *filename*

> or

> > XRECEIVE *wildcard*

> If *options* are specified, they override the global options set in Transfer/Transfer Options for this transfer session only. Options start with "-" or "/". For possible options, see KRECEIVE.

> If "/S" is included in the command, a result code is sent to the host system (or macro). The result code is always 2 characters followed by a Return. For possible result codes, see XRECEIVE.

```
XSEND [options] filespec
XSEND/S [options] filespec
```

> Sends one or more files to a host system using the Xmodem protocol. The *filespec* can be either a single filename or a wildcard.

> If *options* are specified, they override the global options set in Transfer/Transfer Options for this transfer session only. Options start with "-" or "/". For a description of available options, see KRECEIVE, above.

> If "/S" is included in the command, a result code is sent to the host system (or macro). The result code is always 2 characters followed by a Return. For possible result codes see KRECEIVE, above.

```
YRECEIVE [options] [filespec]]
YRECEIVE/S [options] [filespec]]
```

> Receives one or more files from the host system, using the
> Ymodem protocol.

> A file transfer is specified as:

> > YRECEIVE *filename* ⏎

> or

> > YRECEIVE *wildcard* ⏎

> If *options* are specified, they override the global options set in
> Transfer/Transfer Options for this transfer session only. Options
> start with "-" or "/". For possible options, see KRECEIVE.

> If "/S" is included in the command, a result code is sent to the
> host system (or macro). The result code is always 2 characters
> followed by a Return. For possible result codes, see
> XRECEIVE.

```
YSEND [options] filespec
YSEND/S [options] filespec
```

> Sends one or more files to a host system, using the Ymodem
> protocol. The *filespec* can be either a single filename or a
> wildcard.

> If *options* are specified, they override the global options set in
> Transfer/Transfer Options for this transfer session only. Options
> start with "-" or "/". For a description of available options, see
> KRECEIVE, above.

> If "/S" is included in the command, a result code is sent to the
> host system (or macro). The result code is always 2 characters
> followed by a Return. For possible result codes see KRECEIVE,
> above.

```
ZRECEIVE AUTO-OFF
ZRECEIVE AUTO-ON
ZRECEIVE [options] [filespec]]
ZRECEIVE/S [options] [filespec]]
```

> Receives one or more files from a host system using the
> Zmodem protocol, or controls Zmodem auto-reception.

➢ The command "RZ" can be used as an alias for "ZRECEIVE".

When Zmodem auto-reception is on, a program running on the host system can send (download) a file to the PC without the user doing anything. This capability can be turned on or off with

```
ZRECEIVE AUTO-ON
```

and

```
ZRECEIVE AUTO-OFF
```

A file transfer is specified as:

```
ZRECEIVE filename 🔳
```

or

```
ZRECEIVE wildcard 🔳
```

If *options* are specified, they override the global options set in Transfer/Transfer Options for this transfer session only. Options start with "-" or "/". For a description of options, see KRECEIVE.

If "/S" is included in the command, a result code is sent to the host system (or macro). The result code is always 2 characters followed by a Return. For possible result codes, see KRECEIVE.

```
ZSEND [options] filespec
ZSEND/S [options] filespec
```

Sends one or more files to a host system using the Zmodem protocol. The *filespec* can be either a single filename or a wildcard.

> ➢ The command "SZ" can be used as an alias for "ZSEND".

If *options* are specified, they override the global options set in Transfer/Transfer Options for this transfer session only. Options start with "-" or "/". For a description of available options, see KRECEIVE, above.

If "/S" is included in the command, a result code is sent to the host system (or macro). The result code is always 2 characters followed by a Return. For possible result codes see KRECEIVE, above.

10   More On Starting Anzio

This section describes various ways of controlling Anzio's behavior when it starts up.

The following sections describe:

- *Where Command Line Parameters Come From*
- *Command Line Parameters & Default Files*
- *Parameters in EXE Name*

10.1   Where Command Line Parameters Come From

Whenever you start Anzio, or any other Windows program, you are telling Windows to execute a particular command. That command might be just the name of the program to execute. However, it might also have additional parameters ("command line parameters") that tell that program how to behave.

But this may happen behind the scenes, so you might not be aware of it. Here we'll explore how this happens, and how you can take advantage of that to alter program operation.

For instance, you will see below that the parameter "/K" tells Anzio to run in "kiosk mode". We'll explain kiosk mode means later; for now let's look at where the "/K" goes.

10.1.1   Desktop Icons

Your installation program probably created an icon on your Windows desktop that starts Anzio. If you right-click on this icon, and select "Properties", Windows will open a dialog box. If the "Shortcut" tab is not showing, click it.

In the "Target" box is the command that gets executed when you click (or double-click) the desktop icon. At this point, it probably contains the entire pathname of the Anzio executable, such as

```
c:\Program Files\anzio15\anzio32.exe
```

(The directory will depend on where you told the installer to install Anzio. Also, if your program is Anzio Lite, it will say "anzio32r.exe".)

Below this box, in the "Start in" box, is the directory that Windows will change to (like executing a "CD" command) before starting the program.

That is, this is the executing program's home directory. Usually this is the same as the directory containing the EXE, as above.

If you edit the "Target" entry, adding a space and some text at the end, you will be adding command line parameters, which will affect how Anzio operates. If you add " /K", resulting in

```
c:\Program Files\anzio15\anzio32.exe /K
```

and then click "OK", then when you click this desktop icon Anzio will run in kiosk mode.

You can also create your own desktop items (right-click on desktop, New, Shortcut), or copy and paste an existing icon to make a copy, and then modify it.

10.1.2   "Start Menu" Items

Items in the Start Menu hierarchy also have "Target" commands, just like desktop icons described above. You can click on "Start", navigate to an item, right-click on "Properties", and go to the "Shortcut" tab. Then you can modify the Target command as described above.

10.1.3   Batch Strings and Other Programs

In batch strings (.BAT files), scripts, and in any other way you might start Anzio, you will be dealing with a command string. Simply modify this command to include the necessary parameters.

10.2   Command Line Parameters & Default Files

You can call the Anzio program with one or more command line parameters, separated by spaces:

```
Anzio32 /param1 /param2 /param3 name …
```

Command line parameters can run a defined key, specify a settings file, connect to a certain host, set communication settings, or restrict Anzio's menu. Strings used in parameters can be enclosed in double quotes, and *must be* if they contain spaces or tabs.

If there are no command line parameters, Anzio will try to load its standard settings file (AnzioWin.def). This file contains all user settings from the previous Anzio session. The settings file can also specify a key definition file to be loaded. Anzio looks for a settings file in up to three places until found:

1.   The starting directory (the "Start In" directory).

2. The parent directory of the starting directory.

3. The directory containing the Anzio program (if it's different from the starting directory).

If there are one or more command line parameters, Anzio looks at each parameter's length and first character:

- NONE or NEW means to start Anzio without in certain ways as described in the next section.
- A single character *X* means "call defined key *X*" – see *Single Character – Start Defined Key*, below.
- Multiple characters that do **not** start with a slash "/" indicate a name. The name can be a settings file name to be loaded, or a telnet or SSH URL with a username and host name. See *Names – Settings File Name or Telnet Host Name*, page 192.
- A parameter starting with a slash sets an Anzio option – these are described in alphabetical order, beginning on page 193.

10.2.1   <u>'NEW' for New Connection</u>

Use the command line parameter NEW to start Anzio without loading any settings file. Anzio will start up just like after initial installation. It will ask for important starting parameters and begin operation.

10.2.2   <u>'NONE' For Clean Startup</u>

Use the command line parameter NONE to start Anzio without loading the settings file **or any key file.** This is the only way to run without a keys file.

10.2.3   <u>Single Character – Start Defined Key</u>

A command line parameter consisting of only one character (not a slash) will start that defined key (macro) upon startup. The key must be a printable character (not a function key), for example:

```
Anzio32 S
```

will start Anzio and load the standard settings file. In the settings file is an entry for a defined key file, which will then be loaded. At this point, the "S" defined key is called.

Specifying a startup macro causes the Login Wizard **not** to run. The startup macro should start by handling the login task, if necessary.

Note that defined key names are case-sensitive – "s" and "S" are two different keys.

You can also define a key with a command line parameter – see */D Define Key*, page 194.

10.2.4   <u>Names – Settings File Name or Telnet Host Name</u>

Any parameter more than one character long that does not start with a slash ("/") is taken to be a settings file name, or a URL. Some examples:

```
MyAnzio.def
telnet://my.host.com
user123:password777@my.host.com:80
```

> ➢ Any environment variable included in a parameter as "${*variable*}" will be replaced by its value. See section 9.3 *Variable Substitution*. An example might be "${ANZ_MY_DOCS}", which will be replaced by the pathname of the "My Documents" folder; this will vary if multiple users are defined on one PC.

A complete name parameter can contain a network address, including a username and password, and a host name and optional port:

```
[urltype][username[:password]@]name[:port]
```

That is, all parts are optional except *name*.

The *urltype* specifies what kind of connection will be made. Possible values are

```
telnet:
telnet://
ssh://
rlogin://
modem://
```

Anzio reads a name parameter as follows:

1. Anzio strips off an initial *urltype* if it exists, and uses it to determine the connection type.

2. Anzio strips a slash, if any.

3. If the parameter contains an at-sign @, Anzio takes the characters before the @ as a *username* and possibly a :*password*. The username and password (if any) are stored for later use by the Anzio Login Wizard.

4. Anzio first tries to use *name* as a settings file name.

   If the file exists, in the current directory, in the parent directory, or in the directory containing the Anzio program, that file is used instead of AnzioWin.def.

5. If there is no file by that *name*, the name is considered to be a host name or IP address, with an optional :*port*. Anzio loads the standard settings file (AnzioWin.def), but then connects to the given host name rather than the host named in the settings file.

> ➢ A special case is a URL representing a modem. Its format is
>   modem://[*username*[:*password*]@]*phonenumber*
>   Anzio will be configured to do a modem dial-up connection to the
>   indicated *phonenumber*.

For example, if you need to connect to three host systems, you can create three Windows icons, each naming a different settings file, which will contain all the settings appropriate to that host system.

For Internet telnet connections, you can set up AnzioWin.def to contain the most common set of parameters. Then when you activate a telnet cross reference in your browser, the browser will start up Anzio, passing the host name as a parameter. Anzio will start up, read AnzioWin.def, and connect to the indicated host name.

> ➢ Note that if you have a telnet host requiring a special setting, you can
>   create a settings file having the same name as the host. Anzio will
>   find and read that settings file, and so will not read the parameter as a
>   host name. Inside the settings file, specify the desired host name,
>   and Anzio will connect to that host.

10.2.4.1 Settings File on Server

It is also possible to have Anzio to load the settings file from an HTTP, HTTPS, or FTP server. This can provide more security and administrator control.

A server-based settings file is indicated by a standard URL, which can contain a network address, including a username and password, and a file name:

        urltype[*username*[:*password*]@]*filename*

That is, all parts are optional except *urltype* and *filename*.

The *urltype* specifies what kind of server the file is on. Possible values are

> http://
> https://
> ftp://

> ➢ The server that contains the settings file does NOT need to be the same as the server to which Anzio is connecting.

When Anzio loads its settings from a server, it will not allow the user to save settings back to the server.

> ➢ It is also possible to have Anzio load its key definition from a ".kys" file on a server. Use the READ command to specify the URL of the key file.

10.2.5   /C Choose Settings File

The parameter

```
/C
```

causes Anzio to prompt the user for the name of a settings file to use. It will search the working directory for all ".DEF" files, and present a dialog to allow you to choose one.

10.2.6   /D Define Key

Use /D to specify a key definition (macro):

```
/Dk string
```

where *k* is the key to be defined (any printable key), and *string* is the definition. The space separating *k* and *string* is optional. The *string* can be in double-quotes, which would be required if it contained spaces or other special characters (such as "<"). You can embed an Escape character as "<Esc>"; you can embed an F6 as "<F6>".

See *Defined Keys and Macros*, page 45, for more information.

You can use this command line option with the "call key" single-character option. For example, you can define the D key (and save it in the default key file) as:

```
F6 DIAL F6 CALL N | |
```

This definition dials the number returned by another defined key N ("CALL N"). You can then specify the number as a command line option by defining key N:

```
anzio32 D /DN555-1212
```

This command first defines N ("555-1212"), then starts Anzio and runs key D.

10.2.7   **/H Hostname**

Use /H to specify a network host system to connect to, overriding the host name in the settings file:

```
/H:hostname
```

You can specify a host port number as well:

```
/H:hostname:port
```

A numeric parameter following the /H option is also interpreted as a port number, for use with a Netscape browser:

```
/H:hostname port
```

10.2.8   **/K Kiosk Mode**

Use /K to put Anzio into *kiosk mode*. In kiosk mode, Anzio limits the options available to the user, such as printing, dialing, or exiting. Kiosk mode is designed for libraries and other places where public users are expected, and for administrator control of client PCs.

When a particular menu item is not shown, its associated shortcut key is also disallowed. Also, a corresponding action, such as from the popup menu or the Windows close box, is also disallowed.

The /K option disallows **all** menu items and the Anzio help information.

Use /K:*nnn* to allow one or more of the following menu items. *nnn* is the sum of the options:

```
 0 = none
 1 = Print Screen
 2 = Eject
 4 = Quit
 8 = Capture to printer
16 = Capture to file
32 = Copy (to clipboard)
64 = Paste
```

```
 128 = Dial
 256 = Hangup
 512 = Help/About
1024 = (no longer used)
2048 = Send File
4096 = Receive File
```

10.2.9   **/T Communication Type**

Anzio uses /T to specify which type of communication to use:

```
/Ts    serial
/Tt    TCP/IP
/Tr    rlogin
/Tp    PicLan
/Tm    modem
/Th    SSH
```

10.2.10 **/V Verbose**

This switch causes Anzio to present various diagnostic information about its establishing its connection to the host. For instance, for a modem connection Anzio will give various status messages, and also tell you at what baud rates the two modems connected. For a telnet connection, Anzio will show the results of telnet option negotiation. Much of this information can also be seen by going to Edit/Advanced Options/Diagnose/Debug Communications.

10.2.11 **/I Initial Command**

This parameter can tell Anzio to run a particular command on the server, immediately after the authentication phase. Syntax can be as simple as:

```
/Istring
```

Once authentication has taken place, the *string* will be sent to the host. Note, though, that a <return> is NOT assumed. You can indicate a <return> by including a pipe symbol. Also, most commands require some embedded spaces, so surrounding quotes are necessary. Thus, the syntax is usually:

```
"/Istring|"
```

> ➤ This parameter can be used successfully ONLY if Anzio is managing the authentication. That means either a) the connection is via SSH, or b) the connection is via telnet and the Login Wizard is used.

10.3   Parameters in EXE Name

Another way to specify parameters is to rename the Anzio executable in a special way. This is especially useful if you want to start Anzio from a link in a web page, where all you can specify is the name of the program.

For instance, you could have a link in a web page that references

```
http://somehost/anzio32(charset=utf8)(host=nimb
us.anzio.com)(term=vt220).exe
```

(all as one line). When you click on that link, if your web browser allows you to execute (as opposed to save) that item, then Anzio will execute; on startup it will analyze its own program name and will set the charset, host, and term parameters.

For this to work, the EXE file on the web server must have the extended file name as shown, in this case

```
anzio32(charset=utf8)(host=nimbus.anzio.com)(te
rm=vt220).exe
```

Each parameter setting is coded in parentheses, in the form "(*variable*=*value*)". Possible settings include:

```
(host=hostname)
(settings=settingsfilename)
(user=username)
(password=yourpassword)
(term=terminaltype)
(kiosk=ON)
(kiosk=value)
(charset=charsetname)
(commtype=commtypename)
```

Part IV   Technical Reference Guide

11   File Transfer Protocols

This section describes the byte-level behavior of Anzio during file transfer: TRANSMIT, RECEIVE, CAPTURE, and passthrough printing.

> ➢ This section is intended for programmers. See *File Transfer*, page 86, for the operator view of file transfer.

Several source programs that demonstrate the host side of file transfer are included with Anzio – see Appendix E, *Distribution Information*, page 235.

The following sections describe:

- *Simple Upload – TRANSMIT*
- *TRANSMIT LONG*
- *TRANSMIT CRC*
- *RECEIVE*
- *RECEIVE WITH CRC*
- *Simple Receive – CAPTURE*
- *CAPTURE LONG*
- *Passthrough Printing*

11.1   Simple Upload – TRANSMIT

The TRANSMIT command (Transfer/Transmit) reads each line/record from the input file, then sends it to the host. Each input line is terminated by a carriage return (ASCII 13).

Your host operating system may have an 80-byte maximum for input lines (ACCEPT). In this case, you can use TRANSMIT LONG (see next section) to break long records into 80-byte chunks.

If LOCK ON is in effect, Anzio uses *handshaking* for each line transmitted. That is, Anzio waits for an unlock, then sends a line and locks itself. This is repeated until the end of the file is reached. The local file is then closed.

11.2   TRANSMIT LONG

The TRANSMIT LONG command sends records of any length to the host, in small pieces. The host must read each piece and reassemble the record on its end – see the "RECV-PC" example programs in Appendix E, *Distribution Information*, page 235.

Anzio reads each line from the PC file and breaks it into one or more pieces of up to 77 bytes. Each piece is sent to the host as one line, with three bytes of record info:

> *nn*Y*dddddddddd*...

where *nn* is two ASCII characters (01-77) giving the number of data bytes *ddddd*. The third character indicates whether this is the last piece of the record, an ASCII "Y" or "N".

11.3   TRANSMIT CRC

TRANSMIT CRC uses a special protocol for error checking, and is intended only for use with the Universal File Transfer (UFT) utility program, formerly available from Rasmussen Software, Inc.

11.4   RECEIVE

Anzio uses incoming codes from the host to RECEIVE a file. Each record sent by the host is bracketed with a start code DC2 and an end code DC4. These ASCII codes are the same codes used to drive a printer "slaved" to a standard terminal.

In contrast, CAPTURE simply puts all incoming characters into the current output file – see *Data Capture*, page 84.

After Anzio sees a DC2, it puts all succeeding characters into the *receive buffer*, until a DC4. The receive buffer is expanded as needed, up to the available dynamic memory.

> ➢ The receive buffer is used by both RECEIVE and CAPTURE, only one of which may be active.

Incoming data is added to the receive buffer until the host sends an "unlock" to Anzio. Then, if an output file is open:

- ▪ Anzio writes the receive buffer to the PC output file, then clears the buffer.
- ▪ Anzio sends a carriage-return CR to the host, indicating that the host can send the next record.

> ➢ The host must send the appropriate end-of-record marker, if any, inside the DC2/DC4 codes. For example, to end up with text lines in a PC file, the host must send CR and LF codes at the end of each record.

11.5   RECEIVE WITH CRC

RECEIVE CRC receives files using a special protocol for error checking, and is intended only for use with Universal File Transfer (UFT) utility program, formerly available from Rasmussen Software, Inc.

11.6   Simple Receive – CAPTURE

CAPTURE stores each incoming line as the line is displayed on the screen. When the cursor moves to the next line, Anzio writes the stored line to the current output file.

11.7   CAPTURE LONG

CAPTURE LONG captures long records from the host, using the piece-wise protocol described with TRANSMIT LONG, above.

11.8   Passthrough Printing

Various terminal types provide *passthrough printing*, where data from the host passes through the terminal to the printer, without being displayed. The host sends a special escape code to start pass-through print mode, then some printer data, and then a termination code. Anzio recognizes these pass-through codes for each supported terminal type.

Anzio supports pass-through printing to either the printer or a file. If there is an open output file (from OPENO), data is sent there. Otherwise, data is sent to the printer set by PRINTER.

> ➢ If the host does not send a termination code, press ALT A to exit pass-through print mode and return to normal display mode.

12   Sending Anzio Commands From The Host

The host can send Anzio commands by using special control codes. Any command or function listed in the reference section can be called from the host.

For example, Anzio includes file transfer programs for the host which use host control to automate file transfer operations. See Appendix E, *Distribution Information*, page 235.

The host sends commands to Anzio enclosed in command-start and command-end codes. When Anzio receives a command start code (hex 1C, octal 34, decimal 28), all characters until the stop code (hex 1D, octal 35, decimal 29) are processed as Anzio commands. For example, if the host sends:

> *hex-1C* OPENO DOWNFILE *hex-1D*

There is NOT a space after the hex-1C. There is NOT a space before the hex-1D.

> ➢ Allowing commands from the host could represent a security risk. For this reason, Anzio provides a security option to control whether commands from the host are to be obeyed. See the section on Advanced Options, Security.

> ➢ Anzio can be told whether received commands should be displayed or not. This is governed by the Show Commands setting, in Edit/Advanced Options/Keys.

Anzio executes the command:

> OPENO DOWNFILE ⏎

The local file DOWNFILE is opened for output. However, if that file already exists, an error message is displayed on the screen for the operator.

In other words, even though a function is called by the host, all displays, error messages, and input will be to and from the PC screen and keyboard.

> ➢ The host has no way to know if the Anzio command OPENO succeeded. Some commands provide an option "/S" for "send result to host".

For example, OPENO/S sends a two-digit result code back to the host, indicating whether it was successful, or the file exists, or another output file is already open. *Commands*, page 111, describes all commands having the /S option.

(UNIX) You can send Anzio commands with the UNIX shell command "echo":

```
echo "\0034OPENO DOWNFILE\0035" ⏎
```

Where "\0034" indicates octal 34 (hex 1C), and "\0035" is octal 23 (hex 1D). 7. Depending on your UNIX system, you may need to add a parameter "-e" to your "echo" command in order to have it translate embedded octal. Or, you may be able to use the "printf" command.

> ➢ Early software from Rasmussen Software used hex-11 and hex-13, respectively, as command start and command end. These can be re-enabled in Edit/Advanced Options.

13   Security Options

Anzio offers two methods of protecting your communication data (including your password) while it moves over a network such as the Internet: SRP and SSH. Both offer *authentication* and *encryption*.

Authentication is the process of the two players proving who they are to the other. This means that you, the user, can be sure that the host system Anzio is connecting to is actually the system you think it is. It also means you the user have to verify to the host system who you are, the equivalent of supplying your login and password. But in these secure systems, your password never goes across the network in cleartext (unencoded).

Encryption means to encode the data. Although encryption is used during the authentication process, the term is used here as separate from that. That is, encryption refers to encode all data (going both ways) *after* authentication has occurred.

The following sections describe:

- *SRP*
- *SSH*

13.1   SRP

SRP was developed at Stanford University. It is an encryption/authentication *option* under the telnet protocol.

SRP support on the host (server) involves replacing the standard telnetd, password, login, and possibly some other programs on the server.

When SRP is enabled on the host, and Anzio is connecting using the telnet protocol, and Anzio is configured to **allow** or **require** SRP authentication (in Edit/Advanced Options/Security), then Anzio and the server will negotiate and establish an SRP connection. Then if encryption is allowed or required, the main communication of the telnet session will be encrypted.

➢ SRP is not widely used. Most administrators now prefer SSH.

13.2   SSH

The SSH protocol is a separate protocol from telnet, using a different socket (normally 22). On the host (server) end, it requires an SSH

daemon ("sshd") to be running. SSH daemons are available from several sources.

In Anzio, you must choose SSH as your means of connection in the Communicate/Setup dialog box. Further options for SSH are available in the Setup button next to "SSH".

In order to prove who it is, the server will send out a "fingerprint". If Anzio on this PC has not connected to this particular server before, Anzio will display this fingerprint and ask you to verify it. Ideally, for the most security, you will have been given this fingerprint information already by some secure means, and you can confirm to Anzio that the fingerprint is correct, meaning you're connected to the right host. If you have NOT been given this information, and you confirm to Anzio, you will in effect be trusting this first connection as being to the correct host.

After you confirm, Anzio stores the fingerprint in the registry, and future connections are confirmed automatically.

After confirmation of the host, you will be prompted for your login and password. These will be encrypted during transmission. Assuming they are correct, the main session will begin, and it will look just like a normal telnet session.

14   Advanced Options

This section describes some of the more advanced settings within Anzio, accessible under Edit/Advanced Options. They are arranged here by tab.

The following sections describe:

- *General*
- *Screen*
- *Keys*
- *Security*
- *Mouse*
- *Print Wizard*

14.1   General
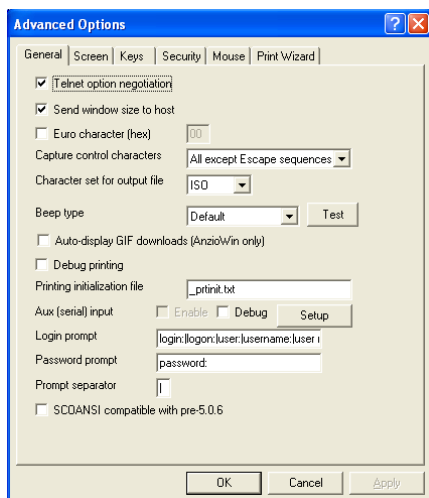
Telnet option negotiation

> When connecting using the telnet protocol, should Anzio negotiate various options with the host? Normally ON.

Send window size to host

> Should Anzio send its window size (rows and columns) to the host, when connected using various protocols such as telnet. Normally ON. Try turning this off if your host system reacts strangely when Anzio switches between 132-column and 80-column modes.

Euro character (hex)
>>Allows you to specify a custom hex value that your host system uses for the Euro symbol (€). This setting takes precedence over Character Set settings. Normally disabled. To enable, check the left box, and enter a hex value in the right box.

Capture control characters
>>When Anzio is capturing incoming characters, this setting specifies how it should treat control characters, including escape sequences. Options are:

>>None   Only plain text is included in the captured data.
>>All   All control characters are captured
>>All except Escape sequences
>>>Escape sequences (such as for screen positioning) are NOT captured.

Character set for output file
>>When Anzio captures to a file, and the incoming data contains non-ASCII characters, this setting specifies by what character set they are stored in the output file. Options are:

>>ISO   The Windows character set.
>>OEM   The OEM (DOS) character set.
>>UTF-8 Unicode UTF-8 form.

Beep type   When a host program sends a code (hex 07) to Anzio to "beep", how should Anzio make a sound? Your choice depends on whether your PC has a sound card, and whether it has speakers. Options are:

>>PC Speaker
>>>Uses the PC motherboard's speaker. This is the only option that will be audible if the PC does not have BOTH a sound card and speakers. Will produce a brief "chirp", that may still be hard to hear.
>>Other settings
>>>Other options are names of Windows sounds. In your Windows configuration (via the Control Panel) you can associate a sound file with each of these names.

Auto-display GIF downloads (AnzioWin only)
> If enabled, then whenever a GIF file is transferred into Anzio from the host (such as with Zmodem), Anzio will display it on the screen.

Debug printing
> If turned on, Anzio will pop up an informational window whenever it prints. This can be useful in diagnosing problems in printing, especially when using AnzioWin's Print Wizard feature. This item is also available in File/Printer Setup.

Printing initialization file
> Tells Anzio the name of a file to be used to initialize the printer, if that file exists. See Print Initialization Files, page 76. This item is also available in File/Printer Setup.

Aux (serial) input
> Allows you to configure and enable input from an "aux" serial device, such as a barcode reader. Items are:
>
> Enable Turns it on
> Debug Causes a debug box to pop up with information on serial data flow
> Setup Allows you to configure the serial connection

Login prompt Tells Anzio's Login Wizard what prompt(s) the host system might display asking for a login name. Multiple entries are separated by the *prompt separator*, below. Default is "login:|logon:|user:|username:" . Case INsensitive.

Password prompt
> Tells Anzio's Login Wizard what prompt(s) the host system might display asking for a password. Multiple entries are separated by the *prompt separator*, below. Default is "password:" . Case INsensitive.

Prompt separator
> What character separates multiple entries in the *Login prompt* and *Password prompt* fields above. Normally the pipe symbol, "|".

SCOANSI compatible with pre-5.0.6
> Ordinarily, Anzio's SCOANSI emulation is compatible with changes introduced in SCO Open Server 5.0.6.

Setting this option causes it to be compatible with earlier versions of that console.

14.2   Screen
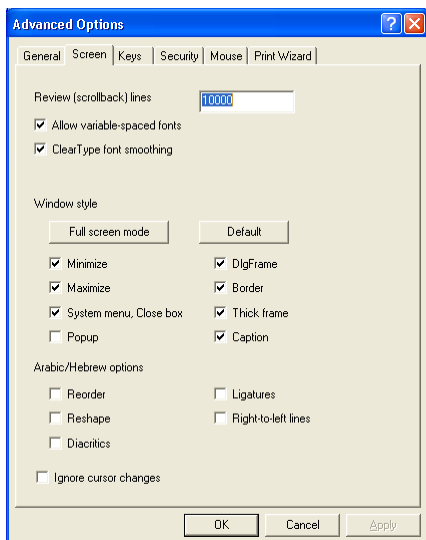
Review (scrollback) lines

Sets the number of lines (at 80 columns) of data that Anzio will store in its scrollback (review) buffer.

Allow variable-spaced fonts

Indicates whether Anzio should present variable-spaced, along with mono-spaced, fonts in View/Screen Font. Generally you'll want to use a mono-spaced font, such as Courier New. However, some Far East fonts are marked as variable-spaced.

ClearType font smoothing

Controls whether Anzio will tell Windows to smooth the edges of the font on the screen, using the ClearType method. Smoothed fonts are arguably more readable. They may cause problems, however, in the alignment of cells' background colors.

Window style

By changing these settings, you can change the appearance of Anzio's window. Some combinations

may not be legal to Windows. Contact us for help if
needed. Options are:

Full screen mode
>   Sets a combination of options to run in a
>   borderless mode, but with the menu bar
>   showing.

Default
>   Sets a combination of options that is the default
>   look of a program window under Windows.

Minimize
>   Should the minimize button be included?

Maximize
>   Should the maximize (zoom) button be
>   included?

System menu
>   Should the "system menu" (attached to the
>   program icon) be included? Also governs the
>   'close' button.

Popup   Sets the window to a "popup" style.

DlgFrame
>   A dialog-style frame has NO menu.

Border   Should the window have a border?

Thick frame
>   Should the window have a thick frame?

Caption
>   Should the window have a title bar?

Arabic/Hebrew options
>   On a Windows system that supports Hebrew and/or
>   Arabic, these items govern whether Anzio should tell
>   Windows to do such things as reordering the text into
>   right-to-left mode, when displaying this data on the
>   screen. If the data coming from the host has already
>   been reordered, so it is in visual order already, then you
>   would want to DISABLE the Reorder setting, for
>   instance.

Ignore cursor changes
>   If checked, then Anzio will ignore escape sequences
>   from the host that would otherwise change the style
>   (block vs. underline, blink vs. not) of the cursor.

14.3   Keys
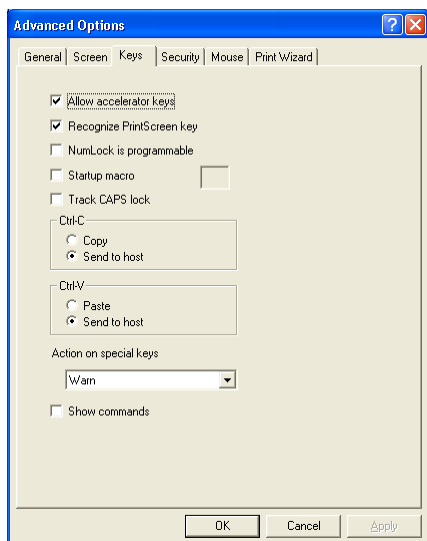
Allow accelerator keys

>> Accelerator keys are those ALT-key combinations that trigger Windows menu events. Normally they are allowed. If you disable this setting, you will be able to use those keystrokes in other ways, such as to assign macros to them. Once assigned, you can turn this back on, because macro assignments have priority over accelerators.

Recognize PrintScreen key

>> It's nice to be able to print the screen by pushing the ⌷PRTSC⌷ key. But you can disable this feature if necessary.

NumLock is programmable

>> If enabled, you can assign a macro to the NumLock (⌷NUM LOCK⌷) key, such as to emulate the PF1 key on a VT keyboard. We don't recommend this, however, because it interferes with normal operation of the NumLock.

Startup macro   You can specify a particular macro key to always run when you start Anzio. Check the left box, and type a key into the right box (normally an alphabetic or numeric key).

Track CAPS lock

>> If enabled, Anzio will remember the state of the Caps

Lock when you switch focus out of Anzio, and will switch back to that state if necessary when you switch focus back to Anzio. This allows you to work in caps mode in Anzio and in non-caps mode in other applications, without constantly switching.

Ctrl-C   You can control whether the control-C key combination causes a copy-to-clipboard, as is the Windows default, or is sent to the host, as is often needed.

Ctrl-V   You can control whether the control-V key combination causes a paste-from-clipboard, as is the Windows default, or is sent to the host, as is sometimes needed.

Action on special keys
This item is for enhanced keyboards, especially those on which the F-keys have dual uses, controlled by an "F-lock" key. If Anzio detects that you have pressed one of these keys, it can warn you that you may need to change the state of your F-lock key. Options are self explanatory.

Show commands
If checked, commands from the host, commands from macros, and their responses to the host will be shown on the screen. Otherwise, they will happen invisibly.

14.4   Security
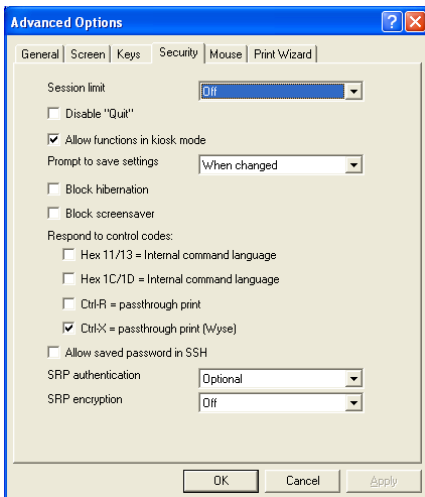
Single instance

> Governs how Anzio will behave if a user starts a second "instance" or session of Anzio. Options are:

> Off   This feature is disabled; user can run multiple sessions.

> Keep existing
>> The new session will be terminated, and the user will be switched back to the already-running session.

> Keep new
>> An already-running session of Anzio will be terminated. User will stay in the new session.

Disable "Quit"   Disables quitting from Anzio at the PC end. This forces the user to terminate (log out of) the host session, which will then cause Anzio to terminate.

Allow functions in kiosk mode

> If Anzio is run in kiosk mode, should macros containing Anzio functions be allowed?

Prompt to save parameters

> When Anzio is terminated, it will generally check to see whether any settings have changed, and if so, ask the user whether they should be saved. This setting can change that behavior. Options are:

> When changed
>> As just described.

> Save without prompting
>> Changed settings will be saved without asking the user first.

> Don't ask, don't save
>> The user will not be asked, and the settings will not be saved.

Block hibernation

> If checked, then Anzio will prevent Windows from going into hibernation mode or other power-saving modes. Doing so can cause problems because it can disrupt a network connection.

Block screensaver

> If checked, Anzio will prevent the Windows screensaver from activating.

Respond to control codes

>> Certain one-byte codes received from the host can cause certain actions in Anzio. However, if received in error, they can cause problems or be a security risk. This section allows you to control them. Codes are:

>>> Hex-11/13 = Internal command language
>>>> In early versions of Anzio, a command sent from the host was bracketed between hex-11 and hex-13 (which are also XON and XOFF). This setting is normally disabled now.

>>> HEX-1C/1D = Internal command language
>>>> The current method for sending Anzio commands from the host is to bracket them with hex-1C and hex-1D. Disable this box for higher security, preventing any Anzio commands to be sent from the host system.

>>> Ctrl-R = passthrough print
>>>> Normally disabled. Control-R is code DC2, which is sometimes used for "aux print".

>>> Ctrl-X = passthrough print (Wyse)
>>>> A control-X starts passthrough print. This applies only when Anzio is emulating Wyse50 or Wyse60.

Allow saved password in SSH

>> When checked, a stored username and password can be used with an SSH connection, just as Login Wizard would otherwise run. Note that this is a weakening of SSH security.

SRP authentication

>> When connecting via telnet, the host system may have the SRP protocol. This setting tells whether SRP authentication (verifying your username and password) is to be done. Options are Off, Optional, and Required.

SRP encryption

>> If an SRP connection is established with the host, and authenticated, should the remaining traffic (that is, the main communication of the session) be encrypted? Options are Off, Optional, and Required.

14.5 Mouse
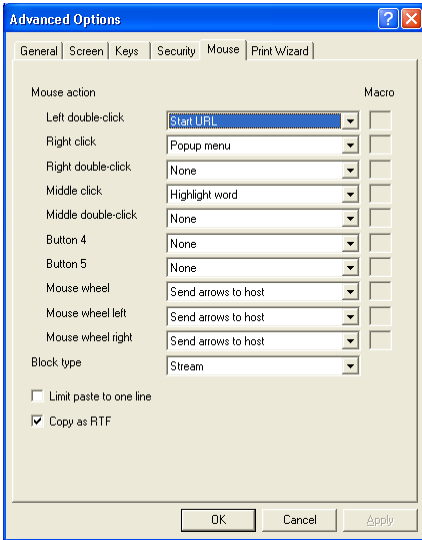
Mouse action (clicks)

Configures the actions Anzio will take on various mouse clicks. Options are:

None   No action
Send word with CR
Sends the "word" pointed to on the screen, with a RETURN.
Send word
Sends the "word" pointed to on the screen.
Execute macro
Execute a user-defined macro, as entered to the right.
Adjust Palette
Adjusts the brightness and contrast of the bitmap shown on the screen, according to the position on the screen where the mouse is clicked. The horizontal axis is contrast, and the vertical is brightness.
Send bracketed word with CR
Picks up a word from the screen, between brackets [ ], strips any leading spaces, and sends it to the host, followed by RETURN

Start URL
>   Picks a string from the screen, and does the equivalent of an Anzio command WINSTART (or a Windows command START) of that string. Usually used to launch a web browser to the URL pointed to on the screen.

Popup menu
>   Pops up a context menu.

Send word (allow underscore), CR
>   Sends the word pointed to on the screen, followed by RETURN. An underscores qualifies as part of a word.

Send word (allow underscore)
>   Sends the word pointed to on the screen. An underscore qualifies as part of a word.

Copy   Copies highlighted text, or entire screen, to the clipboard

Paste   Pastes text from the clipboard to the host

Copy and paste
>   Copies highlighted text (or the whole screen) to the clipboard, and then pastes that text to the host

Character properties
>   Pops up a dialog box giving information on the character being pointed to, and showing an enlarged version of that character

Highlight word
>   Highlights the word pointed to by the mouse

Copy word
>   Copies to the clipboard the word being pointed to by the mouse

---

➢ The characters considered as separators, that is, not as part of a "word", can be configured by editing the settings file.

Mouse action (mouse wheel)
>   Tells Anzio what to do when the user scrolls the mouse wheel. Options are:

None   Does nothing.

Scrollback

> Scrolling up will take Anzio into scrollback (review) mode, where you can see information that has scrolled off the top. In scrollback mode, the mouse wheel can move up and down (regardless of this setting).

Send arrows to host

> Sends up-arrow or down-arrow codes to the host.

Block type
Sets what shape block (or highlight) will be selected if you drag the left mouse button. Options are:

Column

> A rectangular block will be selected. In this mode, when you copy a selection to the Windows clipboard, a bitmap will be copied as well as text.

Stream

> Similar to selection in a word-processing program, but each line (or part) is sent with a RETURN.

Wrap Similar to selection in a word processing program. Lines are assumed to wrap from one to another with one intervening space, so the contents of the highlight are copied as one line.

Limit paste to one line

> If enabled, then when Anzio is pasting information from the clipboard to the host, it will be limited to pasting one line.

Copy as RTF
If enabled, then when Anzio is copying text to the clipboard, it will do so in Rich Text Format (RTF), in addition to doing so as plain text. This allows you to paste into Word, for instance, with fonts and colors.

14.6　　Print Wizard
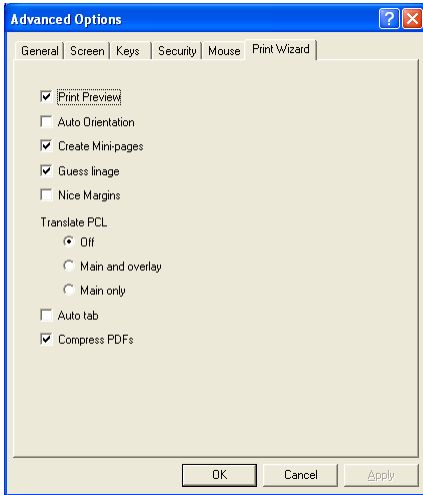
Settings in this tab have an effect only in AnzioWin, and only when printing in Print Wizard mode.

Print Preview

If checked, Print Wizard will present a Print Preview window before printing a job. This lets you see how the data will lay out on the page, and also adjust various margins, etc.

Auto Orientation

If checked, Print Wizard will assume that print jobs should print in portrait orientation, but if they are wide format (more than 99 characters per line), it will switch to landscape.

Create Mini-pages

If checked, Print Wizard will try to determine when a print job is intended for a short page, such as for a receipt or a single label printed on a continuous-feed printer, and automatically create a page of the appropriate length.

Guess linage

If a print job does not contain formfeed characters, Print Wizard must determine how many lines per page to assume. If this item is checked, it will attempt to guess the correct number of lines per page by using pattern analysis; otherwise, it will assume 66 lines per page.

Nice Margins

> If this item is checked, Print Wizard will add up to 1/2"
> margins on the page, if they will fit given the amount of
> text to be printed. If **not** checked, Print Wizard will print
> the data as high and as far to the left as possible for the
> given printer.

Translate PCL

> When a print job contains escape codes in the PCL-5
> language, Print Wizard can either translate them or pass
> them on to the printer (which must be able to understand
> PCL-5). This setting governs how Print Wizard will
> handle such print jobs, and also overlay files that contain
> PCL-5. Options are:

> Off   PCL print jobs will be sent transparently to the
> printer. PCL overlay files will be converted to
> overlay macros and sent to the printer.

> Main and overlay
> Print Wizard will translate the PCL-5 contained
> in both the print job and any overlay file with a
> PCL extension. This means these jobs can be
> printed on any printer; it does not have to be
> PCL compatible.

> Main only
> Print Wizard will translate the PCL-5 contained
> in the main print job. However, if a PCL overlay
> is invoked, Print Wizard will convert it to macros
> and use the printer's macro processing to print
> these macros. This may be faster, but requires
> a PCL printer.

Auto tab   If enabled, then Print Wizard will assume that tab
characters embedded in the print job are separators
between columns; that is, that the input is "tab-separated
values". It will print this data in columnar format.

If DISabled, Print Wizard will assume that tab characters
represent a tab stop at every 8th column.

Compress PDFs

> If enabled, PDF files created by Print Wizard will be
> compressed.

---

15   Customizing Anzio

Anzio can be configured by the user for different hosts and communication environments. You can also customize the Anzio program itself:

- See *Where Command Line Parameters Come From*, page 189, to start Anzio with a particular host, or to set *kiosk* (public use) mode, or to start or define a key macro, etc.
- To change the menus and menu items, use a Windows *resource editor* on the Anzio executable file (anzio32.exe or anzio32r.exe). You can also change or remove accelerator keys, such as ALT X to exit.
- You can edit the settings (.DEF) file with NOTEPAD, for instance.

The following sections describe:

- *Editing the Resources*
- *Editing the DEF File*

15.1   Editing the Resources

A Windows program (or DLL) contains *resources*, which are various kinds of data that can be accessed after compilation. One example of a resource is a menu; another is an accelerator. By using the proper software, you can add, modify, or remove these resources.

The proper software is usually one of the development systems in Visual Studio, from Microsoft. For instance, start up Visual C++. Do File/Open, and navigate to the Anzio executable. Be sure to set "Open As" to "Resources". Then click "Open". This will give you access to the resources of the executable.

If you wish to remove menu items, navigate to "MAINMENU" in the Menu folder, and double-click it. You can then add, change or delete menu items.

> ➢ If you are not running Anzio in English, your resources will be pulled from a DLL instead of from the Anzio EXE. Open the appropriate DLL, such as AnzioFra.DLL for French.

Following are some specific customizations you might consider:

String 2   Alternate title string.

String 3   Alternate help file name.

String 4   Alternative to "anziowin.def" as the default settings file name.

String 5   The ONLY settings file Anzio is allowed to use.

String 6   Alternate default values for settings, in the format of the settings file. Use "\n" for newline, and use "\\" for "\". This can be a partial settings file, as little as one entry. For instance:

```
[Terminal]\nterm=vt320
```

String 7   Required values for some or all settings, in the format of the settings file. Use "\n" for newline, and use "\\" for "\". For instance:

```
[Comm]\nhostname=myserver.com
```

Menu items with ID from 10048 to 10122

If you add a menu item with an ID in this range, then when the user clicks that menu item, Anzio will subtract 10000 from the number, treat it as a decimal character value, and invoke a macro assigned to that key. For instance, 10122 would result in 122, which is the code for "z", so Anzio would invoke macro "z" if it is defined. This allows you to add a custom menu item that corresponds to a macro.

Menu items with ID from 11000 to 11999

If you add a menu item with an ID in this range, then when the user clicks that menu item, Anzio will look for a String resource with the same number. If found, Anzio will execute that string as an Anzio command. So you'll want to add a string of that number also. This allows you to add a custom menu item that corresponds to an Anzio command.

Bitmap "SPLASHALWAYS"

A bitmap that will always be presented as a splash screen, for 5 seconds at program startup.

15.2   Editing the DEF File

Anzio's settings file, such as ANZIOWIN.DEF, is stored in text format. It can be edited using a text editor such as Notepad. Normally, there is not a need to do this. However, some very seldom-used settings are

accessible only in this way; they are not accessible through Anzio itself. Also, certain settings can be used to customize or "lock down" Anzio.

The settings file is divided into sections; section names are indicated by brackets, such as "[Terminal]". Each item below indicates what section it is in.

Following are those items:

Allow-pen-input ([Screen] section)
> This item applies only to the TabletPC platform. If "1" (the default), Anzio will enable pen input if the tablet stylus is moved near the cursor. Set this to "0" to disable this feature.

Ask-1c-command ([Security] section)
> If "1", when a command is received from the host, and commands (starting with 1C) are disabled, user is asked whether to allow commands from host. If "0", user will not be asked and the command will be ignored silently.

Cursor-color ([Screen] section)
> Sets the color of Anzio's cursor, as a decimal value.

C332-pages ([Terminal] section)
> The number of concurrent sessions that can be opened in C332 emulation mode.

SFTP-timeout ([FileTransfer] section)
> The number of seconds to wait when expecting a response in SFTP.

Connecttime ([Network] section)
> The number of seconds to keep retrying for a network connection. Setting this higher allows time for a dial-up modem to establish a connection, for instance.

Connecttime-flags ([Network] section)
> Setting this to "1" tells Anzio to use the "connecttime" operation (above) for ALL network errors, not just for typical ones.

Copy-full-screen ([Mouse] section)
> This setting governs how Anzio behaves if you do a COPY (to clipboard) and nothing is highlighted. If "1" (the default), Anzio will copy the whole screen. If "0", nothing will happen; this can prevent errors in pasting

large amounts of data to the screen. This setting does NOT affect the CLIP macro command.

Decompress-http ([Network] section)
> If "1" (the default), when Anzio fetches a file using HTTP, it indicates that it can decompress files. If "0", Anzio says it can not do this.

Flush-capture ([FileTransfer] section)
> Setting this to "1" tells Anzio to immediately flush to disk anything captured during a CAPTURE FILE operation.

Flush-spooler ([Print] section)
> Setting this to "1" tells Anzio to always flush output when printing in Spooler mode.

Flush-raw-print ([Print] section)
> Setting this to "1" tells Anzio to always flush output when printing in Raw mode.

Font-alt ([Screen] section)
> A string of font names, separated by commas, telling Anzio which alternate fonts to use for the screen, and in which order. Usually ends in "auto", meaning to try all other fonts. Tell Anzio to NOT use a particular font by placing a minus sign in front of its name.

Internet-open-flags ([Network] section)
> Controls how Anzio fetches files using HTTP and FTP. Possible values are $10000000 for asynchronous, $01000000 to allow cache. Flags may be combined.

Internet-open-type ([Network] section)
> Controls how Anzio opens an HTTP or FTP connection. "0" means to use the registry configuration. "1" means direct to the net. "3" means via named proxy (not yet supported). "4" means to prevent Java, scripting, and INS.

Kiosk-mode ([Security] section)
> Setting kiosk mode here, as compared to setting it in the command line parameters, may be considered more tamper-proof. Kiosk mode causes Anzio to run with a restricted menu set. Instead of the full menu set, there is a "smallmenu" menu set (as defined in the resource part of the Anzio executable). All, some, or none of the smallmenu can be displayed.

If kiosk-mode is set to "-1", kiosk mode is disabled, and Anzio runs with its normal menu set. If kiosk-mode is set to "0", Anzio will run with no menu at all. Any other numeric value will be treated as the sum of desired menu items, where each menu item has a number, as follows:

```
   0 = none
   1 = Print Screen
   2 = Eject
   4 = Quit
   8 = Capture to printer
  16 = Capture to file
  32 = Copy (to clipboard)
  64 = Paste
 128 = Dial
 256 = Hangup
 512 = Help/About
1024 = (no longer used)
2048 = Send File
4096 = Receive File
```

Menu-deletions ([Security] section)

Specifies a comma-separated list of menu items to be deleted from the menu at startup, identified by number. For each number, if it is greater than 100, it represents a menu ID. Obtain menu IDs from Rasmussen Software, or by browsing the EXE with a resource editor. If the number is in the form "a.b" (with a period), then a is the menu pulldown column, and b is the item by position (starting at 1) in that column, which can be an active item, a submenu, or a separator.

Xlate-keystrokes ([Terminal] section)

Set to "0" to tell Anzio NOT to translate keystrokes when sending them to the host, such as when using a Far East add-on program. Otherwise leave this line out. See next item.

Utf8-kbd ([Terminal] section)

Set to "2" to tell Anzio that a Far East add-on program is sending it keystrokes in UTF-8. Set to "0", and set

"xlate-keystrokes" to "0" if an add-on is sending keystrokes already in the character set of the host system, such as Big-5. Otherwise leave this line out.

Word-separators ([Mouse] section)

A string of characters that will be recognized as separating words, for the mouse actions that pick up a word from the screen. Note that a space character at the beginning or end of this string in the settings file will be lost, so the space character must be in the middle of the string.

Wrap ([Terminal] section)

This items controls end-of-line wrap for some terminal types in which this must be configured. "1" means wrap, "0" means no wrap.

Voice-modem ([Comm] section)

The Windows name of the modem to be used for placing voice calls. The default is to use the first (or only) modem that Windows lists.

Auto-lf-receive ([Terminal] section)

Governs whether a Linefeed should be automatically added to each Return received from the host. Note that this has been separated from the auto-linefeed for keystrokes, which is accessible in the Communicate menu.

By-page-color ([Terminal] section)

Governs whether Anzio will maintain a separate screen color map for each "page" or "session" of a multiple-page terminal type, such as C332.

By-page-cursor ([Terminal] section)

Governs whether Anzio will maintain a separate cursor style for each "page" or "session" of a multiple-page terminal type, such as C332.

By-page-charset ([Terminal] section)

Governs whether Anzio will maintain a separate character set for each "page" or "session" of a multiple-page terminal type, such as C332.

Decimal-is-period ([Terminal] section)

If set to "1", the decimal key on the numeric pad always

generates a period, regardless of the Windows keyboard configuration.

Ink-cursor ([Screen] section)

Sets the kind of cursor used for ink markup. Default is "2", for a cross-hair cursor. Other options are as defined by Microsoft.

Scanfile-name ([Transfer] section)

The name of the file that will be created on the host if a user does a scan-and-upload from the toolbar, or the command "SENDFILE SCAN://". Note that the extension of this filename also establishes the file format. The default name is "scanfile.tif".

Fax-server ([Printwiz] section)

If this is not blank, it indicates the Windows name of a PC to be used as a fax server, for Anzio's fax output that uses the "Fax32" method.

Allow-ssh-password ([Security] section

If set to "0", Anzio will not do SSH authentication using the password method. Other methods, such as keyboard-interactive, are still available.

SSH-forward ([Security] section)

This must be set to "1" to allow SSH port forwarding.

PCL-typeface-*n* ([PCL] section)

Sets the list of font options for AnzioWin's Print Wizard to try when translating PCL-5 input. There are multiple entries in this style, each with a different *n*, where *n* is a typeface number that can be used in PCL escape codes.

Symbol-set ([PCL] section)

Sets the assumed character set for PCL-5 data that Anzio translates, for cases where the input file does not specify the character set (symbol set). Default is "8U", corresponding to Roman-8.

Typeface-variable ([PCL] section)

Sets the assumed typeface for variable-pitch text in PCL-5 data that Anzio translates. Default is 16602, corresponding to Arial.

Typeface-fixed ([PCL] section)

Sets the assumed typeface for fixed-pitch (monospaced)

text in PCL-5 data that Anzio translates. Default is 4099, corresponding to Courier or Courier New.

Part V   Appendices

Appendix A Error Messages

Anzio displays error messages from two sources: hardware and software. Hardware errors, such as a PARITY error, are displayed on the status line. Software errors, such as "File Not Found", are displayed on the status line or in popup dialog boxes.

This section describes both types of error messages:

- *Communication Chip Errors*
- *Software and File Error Messages*

A.1 Communication Chip Errors

If Anzio detects an error in the PC communications hardware, the error type (PARITY, FRAMING, or OVERRUN) is displayed on the status line.

PARITY   A parity error has occurred. Verify that your PARITY and BAUD settings match the host's.

Some parity errors are caused by telephone line noise over a modem. You may be able to install an error-correcting modem in your PC.

FRAMING   The PC is not receiving the correct number of bits per character. If this error is intermittent, it may be the result of phone noise, as described above. Verify that your BAUD, PARITY, DATA BITS, and STOP BITS are set to match the host's.

Anzio's DATA BITS is the number of bits for data only. DATA BITS 7 and PARITY OFF is a seven-bit character, but DATA BITS 7 and PARITY ON is an 8-bit character.

OVERRUN   When a character arrives at the PC's serial port, it generates an *interrupt*. The PC interrupts whatever it is doing, fetches the character, and puts it into Anzio's buffer. An overrun is caused when another character comes in before the first is processed.

This is almost always caused by poor hardware design choices. The chip that does most of the work in a serial port is called a UART. For serial communication to function well under Windows, at any but the lowest baud rates, the UART must be a type 16550 or equivalent. Earlier chips (8250, 16450) do not have enough

buffering. If you get OVERRUN messages, ask your hardware supplier to determine what kind of UART you have, or contact us for assistance.

Some PCs that have a serial port as part of an integrated chipset, on the motherboard, claim to have a 16550, but don't behave that way. The best solution is to add another serial port, either inside the computer case or a USB-to-serial adapter.

A.2 Software and File Error Messages

This section provides a list of error messages in alphabetical order.

AN INPUT FILE IS OPEN

There is an input file already open. Use CLOSEI to close the current input file, then open the new input file.

AN OUTPUT FILE IS OPEN

There is an output file already open. Use CLOSEO to close the current output file, then open the new output file.

BAD I/O STATUS: *nn* ON *filename*

A file operation returned an unexpected status.

BAD MODEM STATUS

The DIAL command received an unexpected code from the modem.

BAD TAB FORMAT

Anzio could not understand the format of your TAB command.

DISK ERROR ON *command*

A disk error occurred while processing *command*.

DISK OR DIRECTORY FULL

There is no more room on the current output disk or directory.

FILE EXISTS

An OPENO command tried to open an existing file – either delete the file or use a different filename.

FILE OR SUBDIRECTORY ERROR

A file or subdirectory name does not exist or is in the wrong format.

FILE NOT FOUND: *filename*

The specified file was not found – verify that *filename* is the correct name.

INSUFFICIENT MEMORY

The dynamic memory space available to Anzio is full. Anzio uses this memory for both defined keys (macros) and received data.

INVALID FUNCTION

A requested function was not recognized by Anzio.

NO OUTPUT FILE OPEN

An operation requires an output file to be open, and there is none – use OPENO, then try again.

OVERFLOW ON RECEIVE

Data for an incoming file caused a communication chip overflow. The received file is incomplete.

RECEIVE BUFFER OVERFLOW

Data for an incoming file caused a receive buffer overflow. The received file is incomplete.

RESTRICTED COMMAND

You are using a restricted version of Anzio, such as Anzio Lite, which does not support the requested command.

UNABLE TO DELETE: *filename*

The indicated file does not exist, and so cannot be deleted.

UNABLE TO INITIALIZE COMMUNICATION

One or more of the current communication parameters is invalid. For example, the BAUD rate may be wrong, or the PORT number does not exist.

UNABLE TO READ KEY FILE: *filename*

The specified key file could not be found or was in the wrong format.

UNABLE TO TRANSMIT

> Anzio is unable to transmit due to wiring, connection, or communication problems.

UNABLE TO WRITE TO OUTPUT FILE

> Either the output file has filled its disk, or an output device (such as a printer) is not ready.

Appendix B Notes On Particular Host Systems

Host-specific requirements are noted throughout this manual. This appendix collects some pointers for certain host operating systems:

- ▪ *UNIX/Linux Hosts*

B.1 UNIX/Linux Hosts

When working with UNIX or Linux, be sure its TERM setting agrees with Anzio's TERM setting. If you are connecting via telnet or SSH, Anzio will transmit to the host its configured terminal type during the connection process. For other types of connection, you may need to configure the terminal type at the host end, or the host may ask you at login.

You can verify the host's setting by using the "echo" command at the Unix shell level:

```
echo $TERM⏎
```

You can set the TERM variable using the "set" command. To change UNIX's setting, tell UNIX:

```
TERM=VT220; export TERM⏎
```

for instance. Note that some application software on UNIX may assume you are using a certain TERM type, regardless of the current setting.

> ➢ There are particular problems with Wyse 60 (and 50) emulations. Some host systems may want a TERM value of "WY60", while others may want "WYSE60".

It is also important that UNIX be configured properly for backspace and "kill". If these are not set correctly, the characters '@' and/or '#' may not be processed correctly (especially during PC-to-UNIX file transfer). To make sure, tell UNIX that "erase" is *backspace* (ctrl-H) and "kill" is *ctrl-U*:

```
stty erase '^h',kill '^u'⏎
```

This command can be made "permanent" by placing it where it will always be executed on startup. For the command to apply to an individual user account, place the above line in the file ".profile" in the user's home directory. For a command that will apply to every user, place the above line in the file "/etc/profile".

Finally, set Anzio for LOCK OFF, FULL DUP, and BACKSPACE 8.

Appendix C Migration From Earlier Releases

Anzio is constantly evolving to meet the needs of our users. If you are installing a new release, please read the README.TXT file on the distribution disk. This file contains specific instructions for migrating and details on release-to-release changes.

We try to maintain as much downward compatibility as possible. Old command forms will usually work, even when they have been super-seded.

By default, Anzio will install in a directory dependent on its version. This means that a newer release of Anzio will not overwrite an older version. You may want to copy settings files (anything with a ".DEF" extension) from the old directory to the new directory, so you can take advantage of your prior configuration. In fact, the Anzio installer will offer to do this for you.

> ➢ Prior to version 15.0, Anzio's settings files were in a different format. Anzio can still read old-format files, but will save settings in its current text format. These new files can NOT be used by old versions of Anzio.

If you have problems with a new release, start Anzio with command line parameter NEW:

> ANZIO32 NEW ⏎

Anzio will start up in its "virgin" state, without reading a settings, just like the first time it was run a particular PC.

Appendix D Serial Communication Problems

If Anzio is connected to the host on a serial line, and it does not want to communicate at all, it can be difficult to determine the cause. Try these suggestions first:

1. Verify that the cable from the host system is plugged into the PC *serial* port (I know it's obvious, but...).

2. Make sure LOCK mode is off with

   `LOCK OFF` ⏎

3. If the status line still shows LOCK, press [ALT] [U] to unlock the keyboard.

4. Try setting Anzio for different BAUD rates.

5. Try setting Anzio for different comm ports.

6. Test the cable from the host: unplug the serial cable from the Anzio PC and try it on a CRT terminal. If the CRT doesn't work either, try a known good cable. If that doesn't work, check the line back to the host, and the host's terminal line settings.

7. Check the jumpers or switches on the PC serial port card. Make sure Windows' comm port setup agrees with the settings of the actual hardware.

8. Test the PC's serial port hardware with a "loopback test":

   ▪ Disconnect the host cable from the PC. Configure the PC port so that it ends in a 25-pin female plug, using a 9-pin to 25-pin adapter and/or a female-to-female adapter as necessary.

   ▪ Bend a small paper clip, and insert one end into hole 2 and one end into hole 3 on the exposed plug.

   ▪ Now, if you type on Anzio's keyboard, the keystrokes should be echoed back to you (if you are in FULL DUP mode), or appear double (HALF DUP). If nothing shows, your serial port is not working properly.

   ▪ If the loopback test fails, you probably have a defective serial port.

➢ If you have not found the problem after trying these suggestions, contact Rasmussen Software for assistance.

Appendix E Distribution Information

Anzio is distributed on CD-ROM or through a web download. The distribution contains a setup program (such as SETUP.EXE) which, when run, installs the Anzio executables and support files, including these types of files:

*.C	File transfer programs, written in C, for a Unix/Linux host system.
*.CAP	AcuCobol *termcap* file containing terminal settings.
*.DLL	Menu language files
*.DOC, *.TXT	Documentation files.
*.EXE	Executable (program) files.
*.HLP	Help information files.
*.KYS	Key definition files for various terminal emulation types. Most *.KYS files have an association *.DOC file. See *Defined Keys and Macros*, page 45.
*.PRT	Printcode files for FilePro to support Print Wizard.
*.TIC	*Terminfo compiler* files for UNIX host-side terminal settings.
*.TRS	Word Perfect support files.
*.UNI	Unicode mappings for various character sets – see *Using Anzio's Language Support*, page 37.

➢ The following list describes all distribution files in more or less alphabetical order, except the self-explanatory UNI (Unicode mapping) and KYS (key definition) files.

➢ See also the installation file README.TXT.

Anzio32.exe	AnzioWin program file.
Anzio32r.exe	Anzio Lite program file.
ANZIO.CAP	A description of the ANZIO terminal type for Acucobol's A_TERMCAP file.

ANZIO.TIC	A UNIX terminfo file defining host terminal type TERM. Used with ANZIO-TIC.KYS.
ANZIO-M.TIC	A UNIX terminfo file defining host terminal type TERM ANZIO-M for a monochrome. Used with ANZIO-TIC.KYS
ANZIO.TRS	A Word Perfect terminal definition file for the ANZIO terminal type.
ANZIO121.TRS	A later Word Perfect terminal definition file for the ANZIO terminal type. Provides better keymapping and more support for international characters.
ANZIOESP.DLL	The Spanish menu translations.
ANZIOFRA.DLL	The French menu translations.
ANZIOITA.DLL	The Italian menu translations.
AnzioTIC.KYS	Defined key file of function key definitions for use with Anzio.TIC and Anzio-M.TIC.
AnzioTIC.DOC	Documentation for AnzioTIC.KYS.
ANZIOWIN.HLP	The help file used by AnzioWin and Anzio Lite.
AT386.KYS	Defined key file for AT386 emulation.
C332.KYS	Defined key file for C332 emulation.
C332E.KYS	Also needed for C332 emulation.
DOWNLOAD	A UNIX shell script for downloading files to Anzio.
HOWTO.TXT	A text file describing how to implement some of the latest features in Anzio.
KERMIT.KYS	A file of function key definitions to work with Word Perfect on a UNIX system, when WPTERM is set for KERMIT.
PRINTWIZ.DOC	A Microsoft Word file containing instructions for AnzioWin's Print Wizard – see *AnzioWin's Print Wizard*, page 73.
PRINTWIZ.PRT	A printcode file for filePro to support Print Wizard, for monochrome printers.
PRINTWZC.PRT	A printcode file for filePro to support Print Wizard, for color printers.
README.TXT	A file of last-minute notes, etc.

RECV-PC.C	A UNIX source program that allows upload of longer records into UNIX.
SEND-PC.C	A C program used to download UNIX text files to a PC – see *UNIX Download Utility (Send-PC.C)*, page 88.
VT100.DOC	A file that explains the keys used in VT100.KYS
VT100.KYS	Sample function keys for VT100 emulation.
VT220.KYS	Sample function keys for VT220 emulation.
VT220.DOC	A file that explains the keys used in VT220.KYS
VT220N.KYS	A key definition file that more closely matches the "native" VT keyboard.
VT220S.KYS	Sample function keys for VT220 emulation. Uses a different approach to translating PC function keys to VT220 function keys than does VT220.KYS.
VT220S.DOC	A file that explains the keys used in VT220S.KYS.
VT-NUM.KYS	Defines the top four keys of the numeric pad to emulate the VT's PF1 through PF4 keys.
VT420PC.KYS	Key mapping for the VT420 with the PC keyboard.
VT8BIT.KYS	Key mapping for the VT, in which the initiator is hex 9B, instead of "ESC [" .
VWPT.KYS	Key mapping for Viewpoint emulation.
WYSE60.KYS	Sample function keys for WYSE60 (and Wyse 50) emulation.
WYSE60.DOC	A file that explains the keys used in WYSE60.KYS
WYSE60N.KYS	Alternate keymap file for Wyse 60 emulation.