TELNET-RSX
Software User's Guide

Version 2.2-5

MAN/TELNETRSXUG–71

First Printing, March 1985
Revised Printing, May 1989
Revised Printing, July 1990
Revised Printing, July 1992
Revised Printing, May 1993

Contents

1 Introduction

1.1 What Is TELNET-RSX?

TELNET-RSX is a software product that provides TCP/IP virtual terminal networking services for PDP-11 computers made by Digital Equipment Corporation (DEC).

1.1.1 Client and Server Support

TELNET-RSX supports both the client (user) and server ends of the TELNET protocol for the RSX-11M-PLUS and MicroRSX operating systems. Client support means that users on a TELNET-RSX system can log in to remote computers. Server support means that users on remote computers can log in to TELNET-RSX systems.

For the RSX-11M operating system, TELNET-RSX supports only the client.

1.1.2 Protocols

TELNET-RSX implements the following protocols:

- Transmission Control Protocol (TCP)
- User Datagram Protocol (UDP)
- Internet Protocol (IP)
- TELNET protocol

The Department of Defense (DoD) has adopted these protocols as standards for use in all packet networks.

1.2 TCPware® for RSX Family of Software Products

TELNET-RSX is part of the TCPware for RSX family of software products. This family also includes TCP-RSX and FTP-RSX, described in the following table.

Product	Purpose
TCP-RSX	Provides the lower level network protocols (TCP and IP), support utilities, and a QIO programming interface.
FTP-RSX	Allows users to transfer files between remote and local hosts. Supports both the client (user) and server ends of the FTP protocol.

TCP-RSX is included with FTP-RSX and TELNET-RSX software products.

1.3 Connecting Dissimilar Systems

TELNET-RSX is ideal for connecting RSX computers to computer systems made by different manufacturers because the TCP/IP and TELNET protocols are widely implemented, as shown in Figure 1-1. Almost all UNIX-based systems support TCP/IP and TELNET.

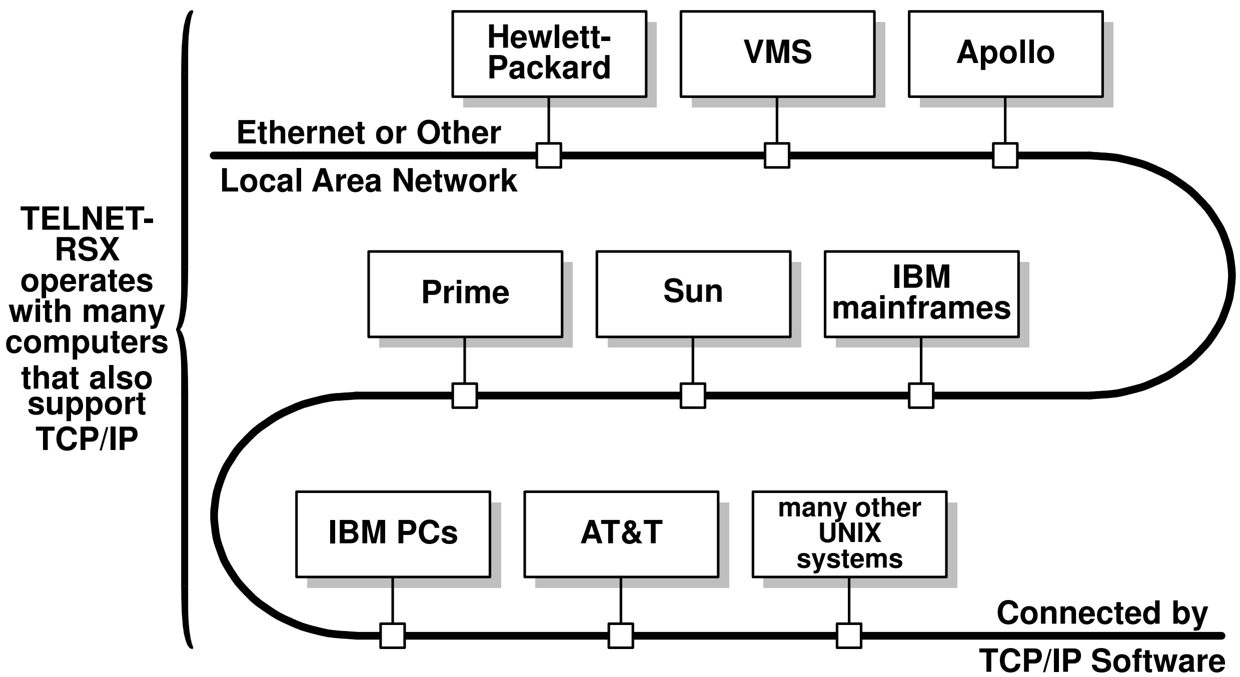

Figure 1-1: Computers That Support TCP/IP

TELNET-RSX also operates with the many network support devices that are compatible with TCP/IP, Ethernet, and other Local Area Networks, as shown in Figure 1-2.

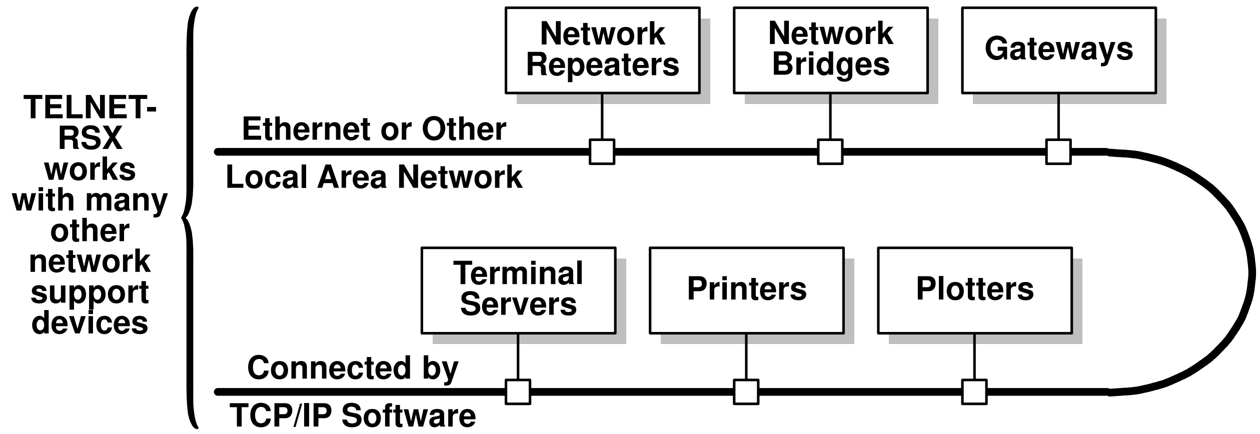

Figure 1-2: Devices that Support TCP/IP

Process Software Corporation develops TCPware for operating systems made by Digital Equipment Corporation (DEC), as shown in Figure 1-3.

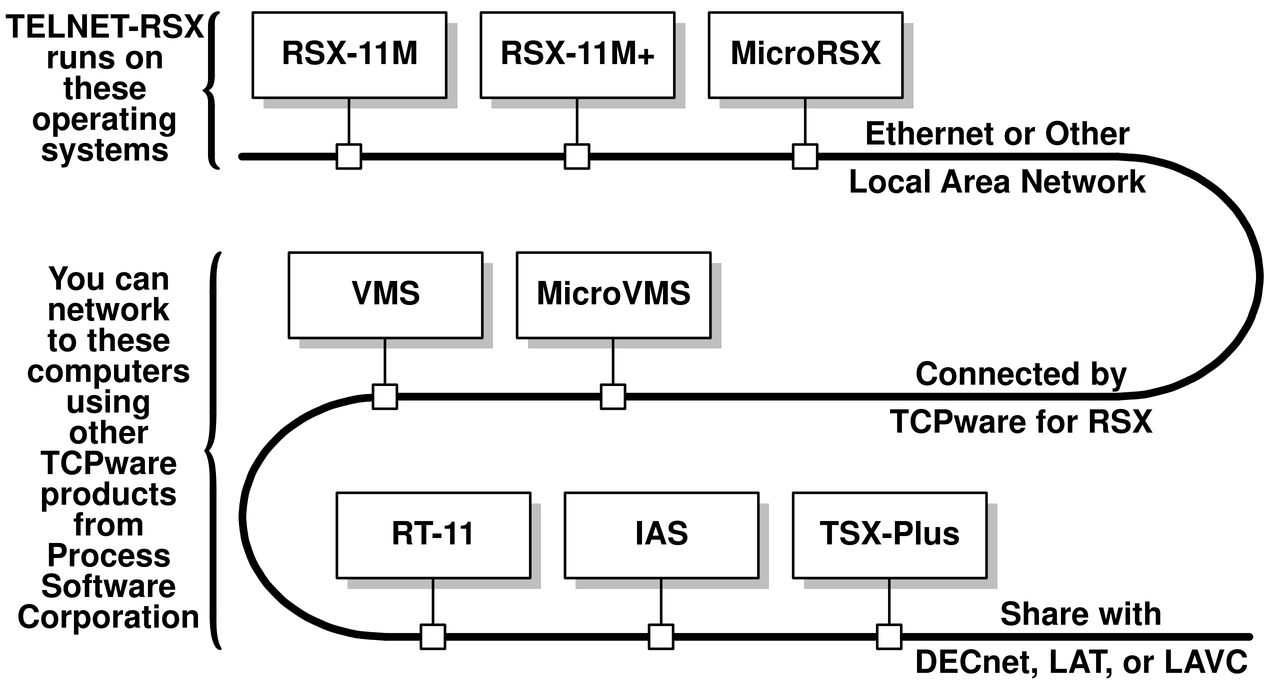

Figure 1-3: TCPware Connecting Different DEC Operating Systems

1.4 Intended Audience

The *TELNET-RSX Software User's Guide* is for:

- system managers who install and maintain TELNET-RSX
- application programmers who use the TELNET-RSX library routines
- users who need virtual terminal networking services

1.5 Prerequisite Knowledge

TELNET-RSX users should be familiar with the DEC RSX-11M, RSX-11M-PLUS, or MicroRSX operating systems. Some familiarity with computer networks is helpful, but not required.

This manual does not document the RSX-11M, RSX-11M-PLUS, or MicroRSX operating systems.

1.6 Contents

The *TELNET-RSX Software User's Guide* contains the following chapters and appendices:

- *Introduction* provides a short overview of TELNET-RSX.
- *Installation,* for system managers, summarizes the steps required to install TELNET-RSX. Complete installation instructions are in the *TCPware for RSX Software Reference Manual*.
- *Using TELNET* shows, step by step, how to use TELNET-RSX to connect to a remote host.
- *General Information* describes the network hardware configuration. It also summarizes the purpose of each utility, TELNET-RSX library routines, and the device drivers.
- *User-TELNET Utility* explains each command provided by the TELNET client.
- *TELNET Library Reference*, for application programmers, describes the TELNET library routines.
- *Appendix A, Options and Control Functions* explains how TELNET-RSX implements the ECHO, SUPPRESS-GO-AHEAD, and BINARY options.
- *Appendix B, Implementation Notes* describes the TELNET implementation restrictions for the RSX environment.
- *Appendix C, Release Notes* summarizes the enhancements for each version of TELNET-RSX.

1.7 The Documentation Set

The complete documentation set for the TCPware for RSX family of software products consists of:

- *TCPware for RSX Software Reference Manual*, which explains how to install and configure TCPware, use the support utilities, and use the QIO programming interface.
- *FTP-RSX Software User's Guide*, which explains how to use the FTP-RSX software product to transfer files between systems.
- *TELNET-RSX Software User's Guide*, which explains how to use the TELNET-RSX software product.

1.8 Conventions

This manual uses the conventions listed below.

`monospaced type`	Monospaced type is used in examples of system output or user input.
UPPERCASE	Uppercase letters are used in commands and examples to indicate that you should enter the characters shown. The system accepts uppercase or lowercase characters.
italics	Italics are used in commands and examples to indicate that either the system supplies a value or you should supply a value.
[ ]	Square brackets indicate the enclosed item is optional. Do not type the brackets when you enter the command.
^	A caret denotes a control character (ASCII characters from 1 to 31). To type a control character, press the “Ctrl” key and type the character that follows the caret.
Host	“Host” refers to any computer system on the network. The local host is the computer you are currently using (your RSX system). A remote host is any other computer system on the network.

1.9 Software License

Each copy of TELNET-RSX includes a *Software License Agreement*. You should read and understand the *Agreement* before installing TELNET-RSX.

The *Agreement* entitles you to install TELNET-RSX on one computer system. If you want to use TELNET-RSX on more than one computer, you must purchase additional TELNET-RSX software products or licenses. Contact Process Software Corporation or your distributor for more information.

1.10 Software Registration Card

A Software Registration Card is enclosed with the TELNET-RSX software product kit. You should fill out and return this card to Process Software Corporation to receive warranty support and software update information.

1.11 Software Update Service

Process Software Corporation periodically updates and enhances TCPware for RSX products to incorporate new features, fix bugs, and improve the documentation. The Software Update Service ensures that you receive the latest versions as they are released. TELNET-RSX includes updates for TCP-RSX. Contact Process Software Corporation to order this service.

2
Installation

This chapter is for system managers. It summarizes the steps required to install TELNET-RSX. Complete installation instructions are in the *TCPware for RSX Software Reference Manual.*

To install TELNET-RSX, perform these steps:

1. **Copy** the distribution kit to a directory on the system.
2. **Build** the components for the product.
3. **Configure** the software.
4. **Get** a software password.
5. **Start** TCPware.
6. **Test** TCPware.
7. **Clean up** after the installation.

3 Using TELNET

This chapter describes, step by step, how to use the User-TELNET utility. It is meant to get you started using TELNET-RSX as quickly as possible.

The *Sample TELNET Session* section in this chapter shows you what a TELNET session is like.

3.1 Preparing for a Remote Session

Before connecting to a remote host, you need to know:

- The name of the remote host that you want to connect to.

  One way to find out this information is to use the Network Control Utility to display the Host Name Table.

  Follow these steps to display the list of hosts on the network:

  1. Run the Network Control Utility by typing:

     `NCU`

  2. At the “NETCU” prompt, type:

     `LIST`

     This displays a list of the names for the hosts on your network. You need not be concerned with the internet addresses.

  3. Exit the Network Control Utility by typing:

     `EXIT`

  The Network Control Utility is described in the *TCPware for RSX Software Reference Manual*.

- The username and password for an account on the remote host. If the remote host does not support multi-user protection features, you may not need a username and password.
- How to use the operating system of the remote host.

3.2 Running TELNET

You connect to a remote host by running the User-TELNET utility, called TELNET.

To run TELNET, type the following at the DCL/MCR prompt:

```
TELNET
```

You should now see the TELNET prompt, “TELNET>”. You can now enter TELNET commands. The next sections describe some of the more commonly used commands.

If the “TELNET>” prompt does not appear, check with your system manager to make sure that the TELNET-RSX software product was installed and started as described in the *TCPware for RSX Software Reference Manual.*

3.3 Starting a Remote Session

Use the TELNET CONNECT command to start a remote session.

Connect to a remote host by typing:

`CONNECT` *host-name*

where *host-name* is the name of the host you want to connect to.

When you connect, a message is displayed that tells you what the escape character is. You should remember the escape character because it is used to return you to the local host.

NOTE

> The escape character (also called the attention character) is used to get the attention of the local host, and should not be confused with the ASCII escape character (27).

For example:

```
CONNECT SYS2
```

connects you to the remote host called “SYS2”.

You now begin the login sequence of the remote host. Once logged in, you interact with the remote host as if your terminal were directly connected to it.

All characters are sent to the remote host, except ^O, ^Q, ^S, and ^X. These characters have special meaning to RSX.

Figure 3-1 shows the relationship between your terminal and the remote host computer during a remote session.

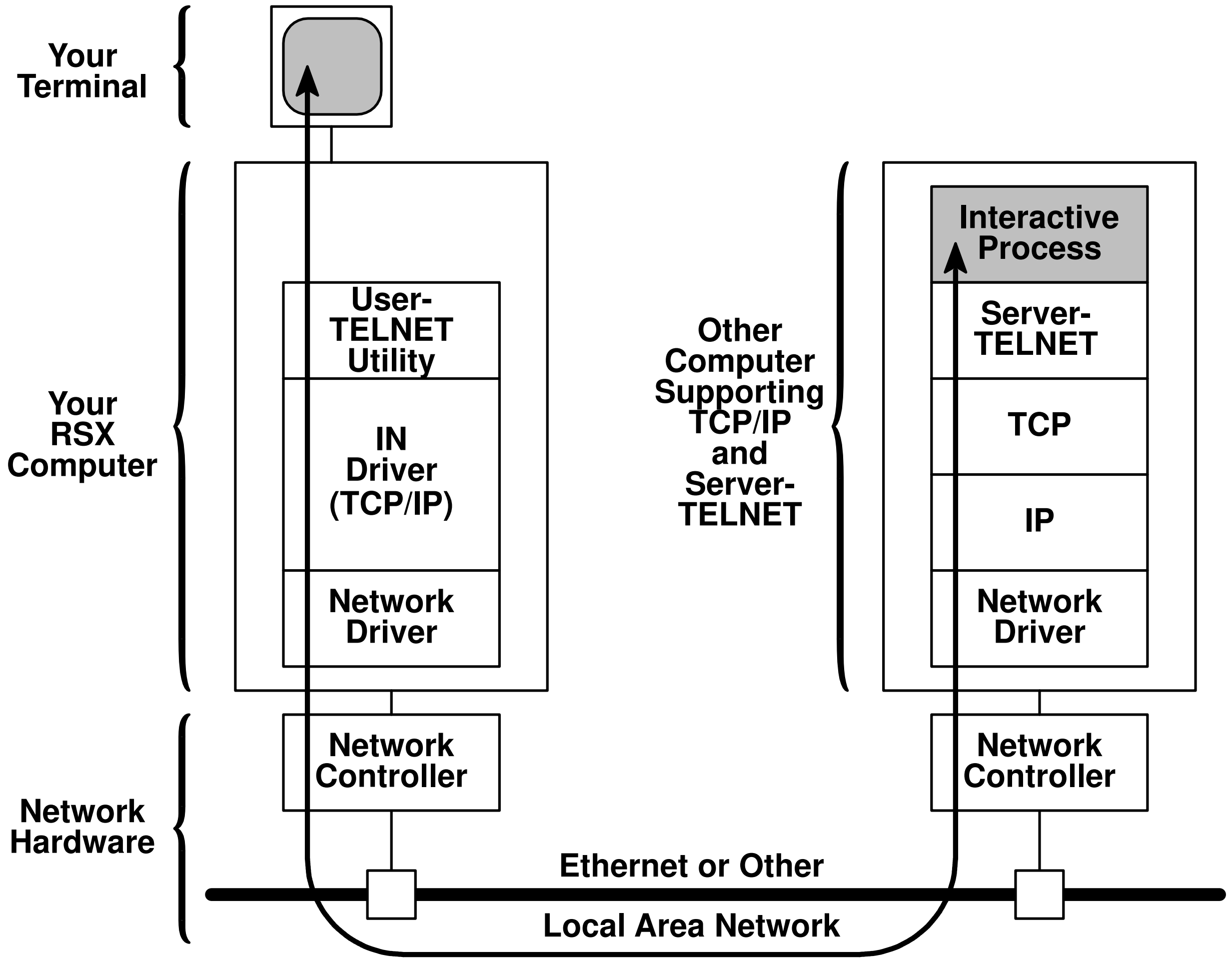

Figure 3-1: RSX User-TELNET Operation

Characters that you type are sent to the remote host. Characters sent from the remote host are received by your terminal. Notice, however, that your terminal is not directly connected to the remote host. Characters are transferred over the network.

3.4 Issuing TELNET Commands During a Session

During a remote session, you may want to issue commands to the User-TELNET utility.

For example, you may want to open a log file to save the output from the remote host in a local file or you may want to change the escape character.

To issue a TELNET command while you are connected to a remote host, type the escape (attention) character. This returns you to the “TELNET>” prompt.

If the escape character is “^\”, you hold down the “Ctrl” key and type the backslash “\” character.

You can now issue a TELNET command. For example:

```
SHOW STATUS
```

This command displays the name of the host you are connected to as well as other information.

See the *User-TELNET Utility* chapter for information on each of the TELNET commands.

Return to the remote host by typing:

```
CONNECT
```

Note that you do not type the host name. This reconnects you to the remote host. Typing a null line by pressing return alone also reconnects you.

You can switch between local TELNET command mode and the remote host as often as you like.

3.5 Ending a Remote Session

Follow these steps to end your remote session:

1. Log out of the remote host. This closes (disconnect) the remote connection and then returns you to the “TELNET>” prompt.
2. Type:

   ```
   EXIT
   ```

   This returns you to the DCL/MCR prompt.

Rather than exiting, you could use the CONNECT command to start a new remote session.

3.6 Additional TELNET Commands

The following are additional commands that you can use while in the User-TELNET utility:

- **DISCONNECT -** to disconnect from a remote host.
- **HELP -** to access the TELNET on-line help facility.
- **SET -** to set characteristics for your present TELNET session.
- **SHOW -** to display information about your present TELNET session.

The *User-TELNET Utility* chapter describes all of the TELNET commands in detail.

3.7 Sample TELNET Session

This section shows a sample TELNET-RSX remote session. In this sample, a user on local host called System1 uses TELNET-RSX to connect to the remote host called System2.

The symbol “^A” means the user typed the escape character. The return key is not explicitly shown at the end of each command line. Vertical ellipses indicate that a portion of the sample is not shown.

You can substitute any commands you like in place of the vertical ellipses. You type the text that is underlined.

```
System1> TELNET
TELNET V2.2  Copyright (c) 1990 Process Software Corporation
TELNET> SHOW STATUS
TELNET V2.2  Copyright (c) 1990 Process Software Corporation

No connection established.

Escape (attention) character is ”^\”.

TELNET> SET ESCAPE ”^A”
?TELNET-I-ESCCHR, escape (attention) character is ”^A”
TELNET> CONNECT SYSTEM2
?TELNET-I-ESCCHR, escape (attention) character is ”^A”
?TELNET-I-TRYING, trying SYSTEM2 (223.42.95.72) ...


Ultrix-32 V3.0 (Rev 64) (system2)
login: smith
Password:
Last login: Tue May 29 11:06:15 from system1
Ultrix-32 V3.0 (Rev 64) System #3: Fri Jun  9 16:54:27 EDT 1989

                Process Software Corporation


Tue May 29 12:51:27 EDT 1990
System2> ls -ax

.         ..        .cshrc    .login    .profile  bin       file1

System2> ^A
TELNET> SHOW STATUS
TELNET V2.2  Copyright (c) 1990 Process Software Corporation

Connected to node SYSTEM2 (223.42.95.72).

Escape (attention) character is ”^A”.
TELNET> CONNECT
   .
   .
   .
   .
System2> ls -ax

.         ..        .cshrc    .login    .profile  bin       file1

System2> logout

TELNET> EXIT
System1>
```

4
General Information

This chapter provides background information on TELNET-RSX, including an overview of the major software components and how they interact.

4.1 Network Hardware Configuration

To operate the software, each computer should be configured with one of the supported network controllers and associated interconnecting hardware and cabling. The hardware components are essentially identical on each computer system.

During most communication operations, the communications network is configured logically as a two node, point-to-point link between the two computers involved in the communication. The network may be physically part of a larger multi-node network with no impact on the other nodes in the network.

The *TCPware for RSX Software Reference Manual* contains more information on the supported network controllers.

4.2 Communications Protocol

TELNET-RSX uses the Transmission Control Protocol/Internet Protocol (TCP/IP) and TELNET Protocol to establish a connection between computer systems. The use of these protocols provides a reliable and efficient means of communication between computer systems.

The Department of Defense (DoD) has adopted the IP, TCP and TELNET protocols as DoD-wide standards for all DoD packet networks. All new DoD packet networks will be using these protocols exclusively.

4.3 Component Overview

The TELNET-RSX software is organized into a number of components. Figure 4-1 shows how they interrelate.

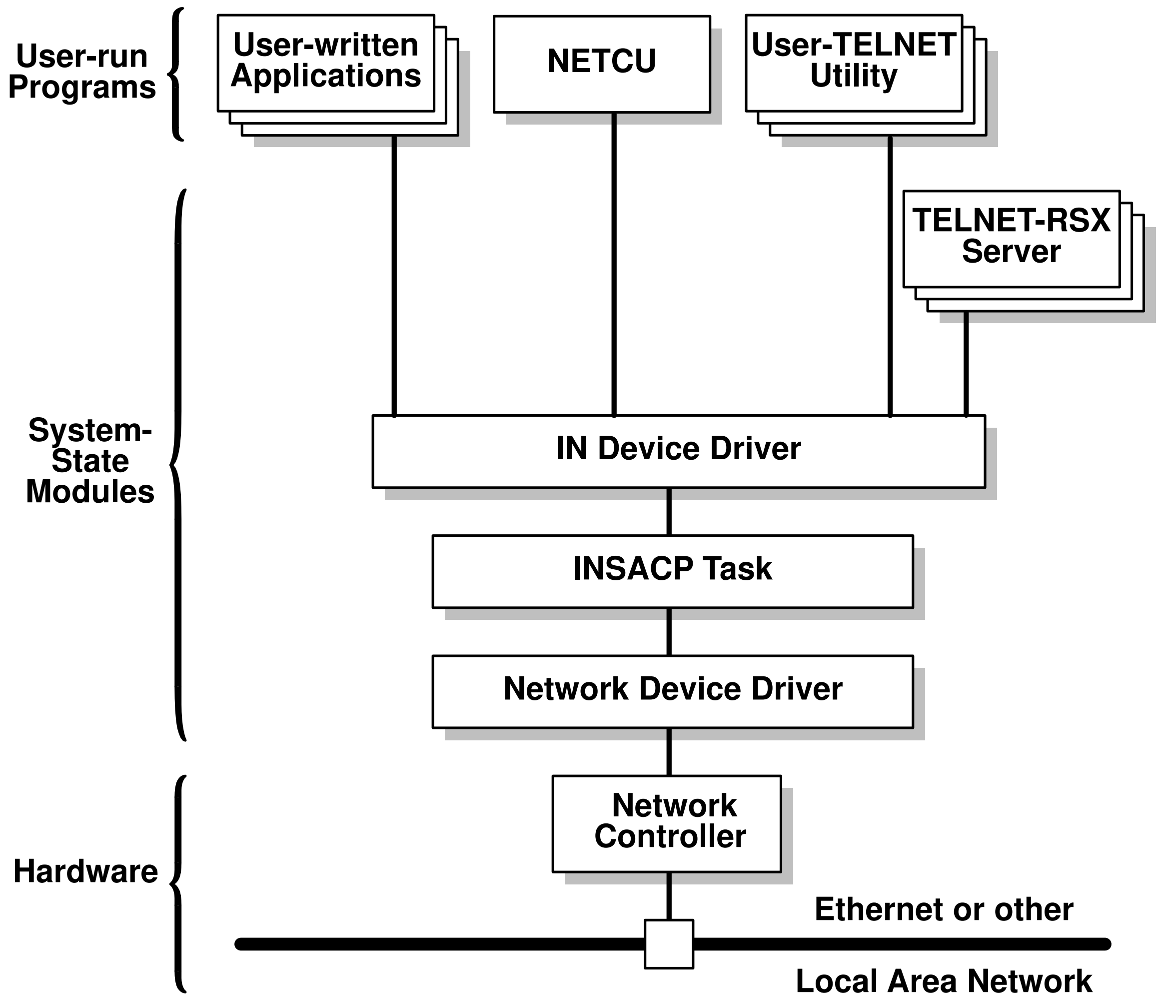

Figure 4-1: TELNET-RSX Components

The purpose and function of each component are described below.

4.3.1 User-TELNET

The User-TELNET utility, called TELNET is the user's interface to TELNET-RSX. You use TELNET to connect to a remote host on your network.

The User-TELNET utility is described in the *Using TELNET* and *User-TELNET Utility* chapters.

4.3.2 Server-TELNET

Server-TELNET provides virtual terminal networking services to remote computers. Users on remote systems can access RSX systems running TELNET-RSX. The remote computer can use any hardware or operating system that implements TCP/IP and the TELNET protocol.

Server-TELNET runs on the RSX-11M-PLUS and MicroRSX operating systems.

Server-TELNET is an RSX TT device driver port interface in the INTELD common. As a result of this implementation, Server-TELNET is completely compatible with the RSX operating system. RSX treats TELNET-RSX terminals in exactly the same way as it treats terminals that are directly connected to RSX.

INTELD manages the Server-TELNET connection. First it listens for a TELNET connection request from a remote host. When INTELD receives a request, it establishes the connection.

INTELD performs TELNET protocol processing, communicating with the interactive process through the TT device driver. The interactive process performs normal terminal input and output. INTELD also transmits data to and from the TCP/IP protocol layers. Each incoming TELNET connection requires one TT unit.

Figure 4-2 shows how INTELD connects remote systems to local RSX systems running Server-TELNET.

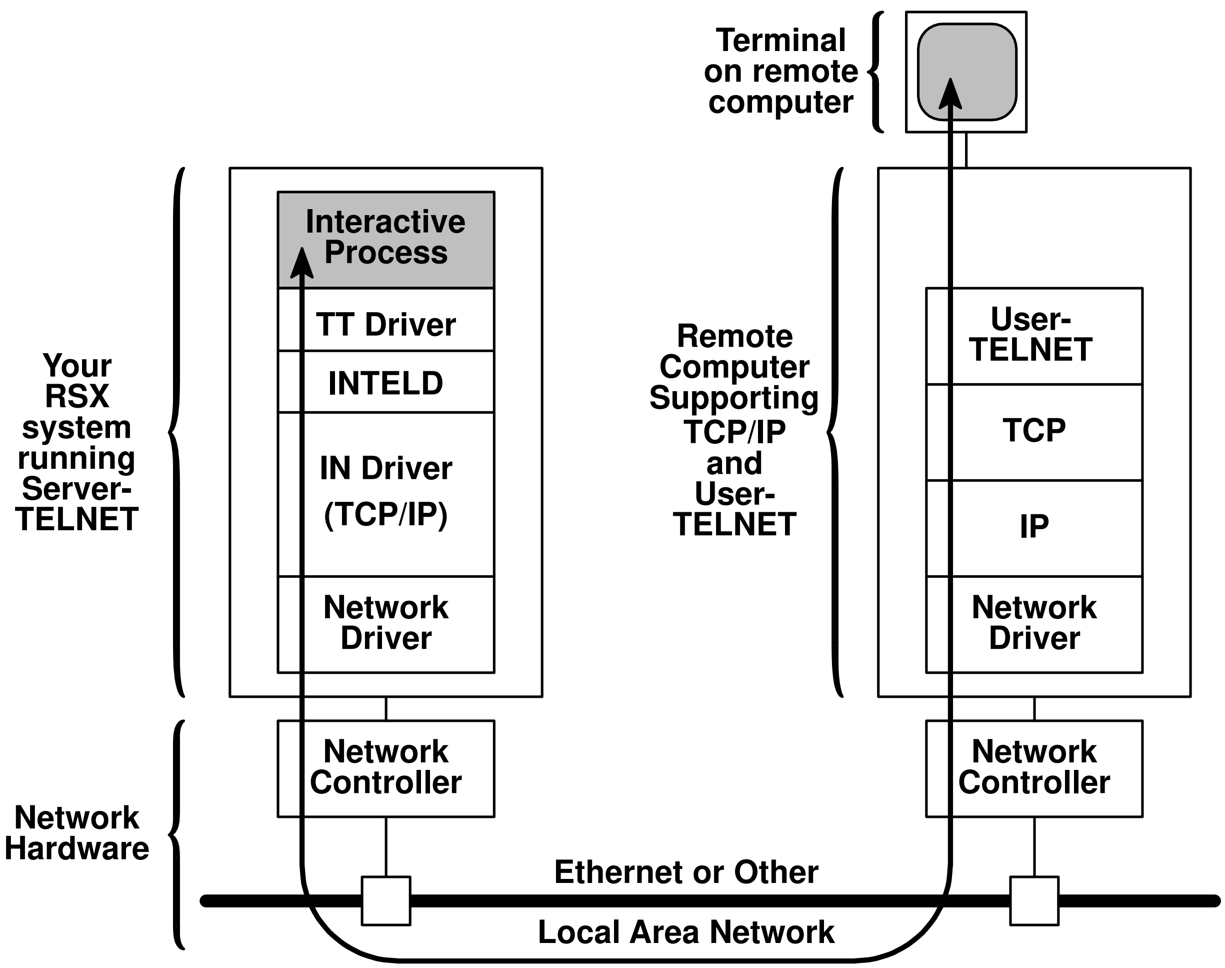

Figure 4-2: Server-TELNET Connecting Remote Terminals to RSX

4.3.3 User-Written Applications

The application programmers can use the user/TCP programming services or the TELNET-RSX library routines. Refer to the *TELNET Library Reference* chapter of this manual and the *TCP Services* chapter of the *TCPware for RSX Software Reference Manual* for more information.

4.3.4 TELNET-RSX Library Routines

The TELNET-RSX library routines are a set of routines that you can link with your programs to perform TELNET operations.

The TELNET-RSX library routines provide a programming interface to the TELNET protocol.

Refer to the *TELNET Library Reference* chapter for information on using the TELNET-RSX library routines.

The User-TELNET utility uses the TELNET-RSX library routines.

4.3.5 IN Device Driver, INSACP, and NETCU

The IN device driver, INSACP, and NETCU all work together to provide the lower protocol layers of the TCP/IP protocol stack (TCP, UDP, IP, ARP).

The IN device driver implements the lower layer protocols. INSACP provides an interface for the IN device driver to the network device and also provides some support functions for IN. The NETCU utility is used to start, stop, and control the networking software.

The IN device driver provides the user/TCP and user/UDP programming services and is used by the User-TELNET utility and your application programs to communicate with other systems on the network using the TCP/IP protocols or UDP protocols.

The TCP and UDP programming services are described in the *IN Device Driver Services* chapter of the *TCPware for RSX Software Reference Manual*. The NETCU utility commands are described in the *Network Control Utility* chapter of the *TCPware for RSX Software Reference Manual*.

4.3.6 Network Device Driver

The network device driver operates the network controller hardware.

The *TCPware for RSX Software Reference Manual* contains more information on the supported network devices.

4.4 References

For more information on the Network Control Utility, the IN device driver, and how to use the IN device driver for task-to-task communications, refer to the *TCPware for RSX Software Reference Manual.*

For more information on FTP-RSX, refer to the *FTP-RSX Software User's Guide*.

For more information on the RSX-11M, RSX-11M-PLUS, and MicroRSX operating systems and utilities, refer to the appropriate DEC documentation.

For more information on the TELNET protocol and TELNET options, refer to the Request for Comments (RFC) documents published by the Network Information Center (NIC) at Government Systems, Inc., 14200 Park Meadow Drive, Suite 200, Chantilly, VA 22021 (800-365-3642, 703-802-4535 (7:00 AM to 7:00 PM EST)). You can also send electronic mail directed to the "HOSTMASTER" and "REGISTRAR" mailboxes at network address NIC.DDN.MIL. The RFCs contain the specifications for all internet protocols.

TELNET-RSX implements the specifications for the protocols documented in the following RFCs:

- *TELNET Protocol Specification*, RFC-854
- *TELNET Binary Transmission*, RFC-856
- *TELNET Echo Option*, RFC-857
- *TELNET Suppress Go Ahead Option*, RFC-858

5
User-TELNET Utility

This chapter describes the operation of the User-TELNET utility, called TELNET.

You run the User-TELNET utility by typing:

```
TELNET
```

The “TELNET>” prompt is displayed. This means that TELNET is ready to accept your commands.

Once you are connected to a remote host, you can return to the TELNET prompt at any time by typing the escape character. The escape character is displayed as you connect to a remote host. Typing the escape character does not log you out of the remote host.

5.1 TELNET Commands

You interact with TELNET by typing commands at the “TELNET>” prompt. TELNET supports the following commands:

- **CONNECT**
- **DISCONNECT**
- **SET ESCAPE**
- **SET LOG**
- **SHOW OPTIONS**
- **SHOW STATUS**
- **HELP**
- **EXIT**

Each TELNET command and its parameters are described in the following sections.

5.1.1 CONNECT Command

Use the CONNECT command to connect to a remote host.

Format

CONNECT [*host-name* [*port-number* [*sec*]]]

If a connection is not presently open, typing CONNECT *host-name* will open a connection to the remote host and log you in.

The connection remains open until you log out of the remote host or enter the DISCONNECT command at the “TELNET>” prompt.

Once a connection is open, typing CONNECT reconnects you to the remote host. The login sequence will not be repeated because you are already logged in.

You can connect to only one host at a time.

You can use OPEN as a synonym for CONNECT.

Parameters

host-name

This is the name of the remote host that you want to connect to. The name can be up to 32 characters in length.

If *host-name* is omitted and a connection is presently open, you are reconnected to that host.

port-number

This is the number of the remote port that you want to connect to. The default port number is 23 (for the Server-TELNET).

sec

This is the number of seconds before the connect command times out. The default is 120 seconds.

5.1.2 DISCONNECT Command

Use the DISCONNECT command to close the current connection. If you are not connected to a remote host, this command has no effect.

Format

DISCONNECT

You can use CLOSE as a synonym for DISCONNECT.

5.1.3 SET ESCAPE Command

Use the SET ESCAPE command to change the escape (attention) character.

Format

SET ESCAPE *char*

This command allows you to change the escape character to a key which is more convenient for you. The default escape character is “^\”.

You will probably want to change the escape character if the remote host uses that character to perform some function or if your terminal can not generate the character.

You can use ESCAPE as a synonym for SET ESCAPE.

The characters ^O, ^Q, ^S, and ^X are not allowed as escape characters.

Parameter

char

This is the character that you want to change the escape character to.

Parameter *char* can be specified in either of the following formats:

- Numeric - The decimal number represents the ASCII value of the character.
- String - The quoted string is the character. Control characters can be specified by preceding the character with a caret (^).

Examples

For example, the following commands set the escape character to control-B (ASCII 2). They are equivalent.

```
SET ESCAPE "^B"
```

```
SET ESCAPE 2
```

The command

```
SET ESCAPE "}"
```

sets the escape character to right brace (}).

5.1.4 SET LOG Command

Use the SET LOG command to open or close a log file. A log file is used to save the output received from a remote host. While connected to a remote host, all output that is sent to your terminal from the remote host is also put in the log file.

Format

SET LOG [*file*]

Opens local file *file* and begins logging.

To close a log file (and stop logging), type SET LOG with no file specification.

If there is an open log file and you type SET LOG *file*, the open file is closed and a new log file is created.

Parameter

file

A local RSX file specification that the remote host's output is logged to.

If *file* is omitted, the present log file (if there is one) is closed.

5.1.5 SHOW OPTIONS Command

Use the SHOW OPTIONS command to display information about the current options in effect.

Options modify the way TELNET handles your terminal over the network. When a connection is first established, both hosts negotiate which options to use based on the options that each host supports.

Format

SHOW OPTIONS

See the *Options and Control Functions* appendix for information on the supported options.

5.1.6 SHOW STATUS Command

Use the SHOW STATUS command to display information about your present TELNET session.

Format

SHOW STATUS

The following information is displayed:

- name of the remote host if a connection is open
- name of the log file if one is open
- present escape character

You can use STATUS as a synonym for SHOW STATUS.

5.1.7 HELP Command

Use the HELP command to display a brief description of each TELNET command on your terminal.

Format

HELP

5.1.8 EXIT Command

Use the EXIT command to exit the User-TELNET utility and return to DCL/MCR.

Format

EXIT

If there is an open connection, it will be disconnected before exiting. If there is an open log file, it will be closed before exiting.

Once you exit, you must run the User-TELNET utility again to connect to a remote host.

You can use QUIT and BYE as synonyms for EXIT.

5.2 TELNET Messages

TELNET-RSX issues informational and error messages. These messages are self-explanatory and conform to the standard RSX message format.

NOTE

> If the IE.ONP (Hardware/Software option not present) error message is issued by TELNET, the software password is invalid for the current configuration. Ask your system manager whether the software password was entered correctly.

6
TELNET Library Reference

This chapter is for programmers. It describes the TELNET-RSX library routines. These routines provide a programming interface to the TELNET protocol. Use these routines in your applications to provide TELNET capabilities.

For complete information on the TELNET protocol, refer to the documents listed in the *General Information* chapter, *References* section.

6.1 List of Routines

The TELNET-RSX library contains the following routines:

- **TELABO** - Abort a TELNET connection.
- **TELCCB** - Initialize a connection control block (CCB).
- **TELCLS** - Close a TELNET connection.
- **TELCMD** - Send TELNET commands.
- **TELGET** - Get the value of a CCB field.
- **TELOPN** - Open a TELNET connection.
- **TELRCV** - Receive data.
- **TELSET** - Set the value of a CCB field.
- **TELSND** - Send data.

These routines are in the TELLIB.OLB object library.

6.2 Connection Control Block

This section describes the connection control block (CCB). The CCB contains all the information required to establish and maintain a TELNET connection. The CCB is a 1078-byte buffer used by the TELNET-RSX library routines.

Your application program allocates storage space for the CCB. When the TELNET-RSX library is not using the CCB, you are free to use it for other purposes. Your application program cannot alter or access a CCB when the the TELNET-RSX library is using the CCB.

The TELNET-RSX library routines reference the CCB via a connection control block identification number (CID). The CID is the starting address of the CCB. Each TELNET-RSX library routine requires the CID parameter.

The TELCCB library routine must initialize the CCB before the TELOPN routine is called. After initialization, the TELNET-RSX library considers the CCB to be in use. The CCB is marked as free when the TELNET connection is fully closed (see TELCLS) or aborted (see TELABO). You must re-initialize the CCB before trying to open another connection with TELOPN.

In the future, the TELNET-RSX library might support multiple connections within the same task. Currently, however, the library supports no more than one active TELNET connection.

The TELSET and the TELGET library routines can access CCB fields. The symbolic names for the CCB fields are defined in the TELNET-RSX object library.

The following is a list of CCB fields and their uses:

- **T$REFN** **-** Unsigned word specifying the receive event flag number to be set when data is received. You can get this field.
- **T$TEFN** **-** Unsigned word specifying the transmit event flag number to be used when sending data over the TELNET connection. You can get this field.
- **T$NLUN** **-** Unsigned word specifying the logical unit number to be used for the network operations. You may get this field.
- **T$CMDR** **-** Unsigned word specifying the starting address of your TELNET command processing routine. You can get and set this field.
- **T$UAST** **-** Unsigned word specifying the starting address of the AST routine to be called when data is received. If the value of this field is zero, then your application program must use the event flag specified in the T$REFN field of the CCB to determine when data is available. You can get and set this field.
- **T$TSIZ** **-** Unsigned word specifying the number of untransmitted bytes in the internal send buffer. This value reflects the amount of raw data in the internal send buffer. This number may be larger than the actual number of user data bytes you have sent due to the requirements of the TELNET protocol. You can get this field.
- **T$RSIZ** **-** Unsigned word specifying the number of unread bytes in the internal receive buffer. This value reflects the amount of raw data in the internal buffer. This number may be larger than the actual number of user data bytes you will read due to the requirements of the TELNET protocol. You can get this field.
- **T$OPEN** **-** Unsigned word used as a flag that indicates the status of the connection. This field is zero if there is *no* connection currently open, otherwise a connection is open. You can get this field.
- **T$RIA** **-** A 4-byte “string” representing the remote internet address for the connection. You can get this field. The internet address is stored in internet byte order. For example, internet address 1.2.3.4 is stored as .BYTE 1,2,3,4.

- **T$TPEN** - Unsigned word specifying the number of transmit ASTs pending. This value is incremented for each transmit issued (queued) and decremented as the corresponding ASTs complete. There is at most one transmit request outstanding. A value greater than one for this field will result if you disable ASTs. If this value is zero then there is no data currently queued for transmission. You can get this field.
- **T$RPEN** - Unsigned word specifying the number of receive ASTs pending. This value is incremented for each receive issued (queued) and decremented as the corresponding ASTs complete. There is at most one receive request outstanding. A value greater than one for this field will result if you disable ASTs. If this value is zero then there is no receive request currently outstanding. You can get this field.
- **T$TQF** - Unsigned word used as a transmit queued flag. If this field is non-zero then there was a transmit request previously issued that is currently awaiting completion or recognition by the TELNET-RSX object library. You can get this field.
- **T$CLSR** - Unsigned word used as a flag which is non-zero if a close has been received (i.e., remote end has issued a close). You can get this value.
- **T$CLSS** - Unsigned word specifying a flag which is non-zero if a close has been sent (i.e., local end has issued a close). You can get this value.

Additional fields are used internally by the TELNET-RSX library routines and are not accessible to your application programs.

6.3 Library Routines

This section describes each of the TELNET-RSX library routines.

All routines use the standard PDP-11 R5 calling convention. All routines should be called as integer functions because a status code is returned.

Status codes are returned to indicate the outcome of the operation. The symbolic names for the status codes are defined in the TELNET-RSX object library TELLIB.OLB. Positive values indicate success; negative values indicate errors.

See the *Status Codes* section later in this chapter.

6.3.1 TELABO Routine

The TELABO library routine is used to abort a TELNET connection.

Format

ret-status = TELABO(*cid*)

Argument

cid

The connection ID (starting address of the CCB).

Description

This routine aborts the TELNET connection. This routine returns when all activity on the connection has completed.

This routine can be called at any time after the CCB has been initialized and when ASTs are enabled. You must call this routine after receiving the SC.CID error code.

This routine marks the CCB as no longer initialized (free). You are free to use the CCB for other purposes.

6.3.2 TELCCB Routine

The TELCCB library routine is used to initialize a connection control block (CCB).

Format

ret-status = TELCCB(*cid,netlun,base-efn,ast-rtn,cmd-prc-rtn*)

Arguments

cid

Connection control block identification number. The *cid* number is the starting address of a 1078-byte buffer to be used for the connection control block. You must use this *cid* number when calling any of the other TELNET-RSX library routines. This *cid* number specifies the starting address of the connection control block for the various TELNET-RSX library routines.

netlun

The logical unit number (LUN) to be used for the TELNET connection. You must never use this LUN.

base-efn

First event flag in a sequence of two event flags. The first event flag (referred to as the receive event flag) is used for data received over the network. The second event flag is used for data transmitted over the network.

You must never set or clear either of these event flags.

The receive event flag is set whenever data is available in the receive buffer. It is cleared when the last byte of data is removed.

ast-rtn

AST routine to be called when data is received over the network. If the *ast-rtn* is zero, you must use the receive event flag for data notification.

cmd-prc-rtn

The command routine that is called each time a TELNET command is received. The *cmd-prc-rtn* is the starting address of your command processing routine.

The inputs to the command processing routine are the CID for the connection, a buffer containing one complete command, and a byte count for the buffer. Please see the *User Command Processing Routine* section for more detailed information on the command processing routine.

Description

This routine initializes the CCB specified by the *cid* argument.

A CCB must be initialized by the TELCCB routine before any of the other TELNET-RSX library routines can be called. Each time a new TELNET connection is to be opened, you must first initialize the connection control block with the TELCCB routine.

You can only call TELCCB when the CCB is “free”. Failure to do this may result in fatal errors. See the *Connection Control Block* section in this chapter for a description of the CCB.

You must not attempt to access the CCB data fields except by using the TELSET and TELGET routines.

6.3.3 TELCLS Routine

The TELCLS library routine is used to close a TELNET connection.

Format

ret-status = TELCLS(*cid*)

Argument

cid

The connection ID (starting address of the CCB).

Description

Call this routine when you have finished sending data to the remote host or when the remote host notifies you that it has closed its end of the connection. ASTs must be enabled when calling TELCLS.

This routine initiates a close on the local end of an open connection. For a connection to be fully closed both ends of the connection (local and remote) must be closed. No more data may be sent or received once a connection is fully closed.

Additional data may be received until the remote end closes its side of the connection. Therefore, you must call the TELRCV routine until the IE.EOF (connection closed) status is returned.

The two typical close situations are detailed below:

1. Remote end closes first. In this case, TELRCV returns the IE.EOF (connection closed) status and then you call TELCLS to close your end of the connection. TELCLS returns when the connection is fully closed.

   In this case, you cannot send or receive any more data.

2. Local end closes first. In this case, you call TELCLS to close your end of the connection. TELCLS returns but the remote end of the connection is still open. You must call TELRCV until it returns the IE.EOF (connection closed) status, at which time the connection is fully closed.

   In this case, you cannot send any more data, but you may receive data that was previously sent by the remote host.

In general, TELRCV must return IE.EOF (connection closed). This can happen before or after you call TELCLS.

Once both ends of the TELNET connection are closed, the CCB is marked as no longer initialized (free) and you are free to use the CCB for other purposes.

6.3.4 TELCMD Routine

The TELCMD routine is used to send TELNET commands to the remote host. Commands include option commands and control functions (such as BREAK and AYT).

Format

ret-status = TELCMD(*cid*,*buffer*,*byte-count*)

Arguments

cid

The connection ID (starting address of the CCB).

buffer

The address of your buffer containing TELNET commands to send to the remote host. The commands must be complete and correct.

byte-count

The length of the buffer in bytes. The *byte-count* argument is the address of a word containing the number of bytes in the command buffer.

Description

This routine is similar to the TELSND routine. TELCMD uses the same internal transmit buffer; however, no processing is done to the command data. The buffer is transmitted with IAC and all other characters left as is.

TELCMD can be called any time after TELOPN has completed successfully, and before the connection is aborted or closed.

6.3.5 TELGET Routine

The TELGET library routine is called to get information from a CCB field.

Format

ret-status = TELGET(*cid*,*field-code*,*value*)

Arguments

cid

The connection ID (starting address of the CCB).

field-code

Identifies the CCB field value you want to get. The *field-code* is the address of a word which specifies which field in the connection control block is to be read. The symbolic definitions for the CCB fields are defined in the TELNET-RSX object library, TELLIB.OLB.

See the *Connection Control Block* section for more information on each field.

value

The returned value of the specified field. The *value* argument is the starting address of the buffer that will receive that value of the CCB field. The length of the buffer depends on the field requested in the *field-code*.

Description

The value of the CCB field you requested is returned to you in the argument *value*.

TELGET can be called any time after TELCCB has been called and before the connection is closed or aborted.

6.3.6 TELOPN Routine

The TELOPN library routine opens a TELNET connection to a remote host.

Format

ret-status = TELOPN(*cid,host-name*,[*port* [,*timeout*]])

Arguments

cid

The connection ID (starting address of the CCB).

host-name

The name of the host you want to connect to. The *host-name* argument is the starting address of an ASCIZ string of the remote host's name (may be in internet address form). The maximum length of the *host-name* argument is 32 characters.

port

Unsigned 16-bit TCP port number to use for the connection. The *port* argument is the address of a word containing the port number. For a normal TELNET connection, the port number is 23.

This argument is supplied for use in special applications where the remote server is not a standard TELNET server.

If this argument is omitted or zero, a default value of 23 is used.

timeout

The time-out value for the connection. The *timeout* argument is the address of an unsigned word containing the number of seconds to wait on the connection before timing out.

If this argument is omitted or zero, a default value of 120 seconds (2 minutes) is used.

Description

This routine opens an active TELNET connection to the remote host (specified in the argument *host-name*). After the TELNET connection is opened, a network read request is initiated.

See the *General Information* chapter in the *TCPware for RSX Software Reference Manual* for more information on internet addresses and host names.

The CCB must be initialized by the TELCCB routine before you call TELOPN.

A connection must be opened before you can send or receive data or commands.

6.3.7 TELRCV Routine

The TELRCV library routine is used to obtain data that has been received from a remote host.

Format

ret-status = TELRCV(*cid*,*bufsize*,*buffer*,*byte-count*)

Arguments

cid

The connection ID (starting address of the CCB).

bufsize

The size of your data buffer in bytes. The *bufsize* argument is the starting address of a word containing the number of bytes you are willing to receive.

buffer

The starting address of your buffer to receive the data.

byte-count

The number of bytes actually returned to you in your buffer. The *byte-count* argument is the address of a word that receives the length of the data returned in your buffer.

Description

This routine should be called only when data is available. You can use the AST (CCB field T$UAST) or the receive event flag (CCB field T$REFN) specified in the call to the TELCCB (or set by the TELSET routine) to determine when data is available.

The receive event flag is set when data has been received and is in an internal receive buffer. You must then call TELRCV to obtain the data/commands received. TELRCV must be called until the receive event flag is clear at which time a network read request is issued.

If you are using an AST routine for data notification, upon receiving the AST you must call TELRCV until the receive event flag is clear or no data is returned.

You should not use this routine to poll for data because it wastes CPU time.

Commands are returned to you via your TELNET Command Processing Routine. See the *User Command Processing Routine* section.

Only TELNET data is returned in the *buffer* argument. A NULL following a carriage return is removed and an IAC IAC is translated to a single IAC. This routine ensures that TELNET commands and data are processed in the order in which they are received. It does this by copying TELNET data to your buffer until your buffer is full, there is no more data, or a TELNET command is encountered. If a TELNET command is encountered and data was copied to your buffer, the command will not be processed until the next call to TELRCV (at which time any additional data following the TELNET command or commands is also returned).

For a connection to be fully closed, both ends must issue and receive a close.

If a close is initiated from your local end, TELRCV must be called until the IE.EOF (connection closed) status is returned. See TELCLS.

If a close is initiated from the remote end, TELRCV returns the IE.EOF (connection closed) status. Upon receiving the connection closed status, you must call TELCLS to fully close the connection. See TELCLS.

TELRCV can be called any time after TELOPN has completed successfully and before the connection is or aborted or has closed.

6.3.8 TELSET Routine

The TELSET library routine is called to set a value of a CCB field.

Format

ret-status = TELSET(*cid,field-code,value*);

Arguments

cid

The connection ID (starting address of the CCB).

field-code

Identifies the CCB field value you want to set. The *field-code* argument is the address of a word containing the symbolic value for the CCB field to be set. The symbolic definitions for the CCB fields are defined in the TELNET-RSX object library, TELLIB.OLB.

See the *Connection Control Block* section for more information on each field.

value

The value of the specified field. The *value* argument is the starting address of the buffer containing the value to set the field to. The length of the buffer depends on the field requested in the *field-code* parameter.

TELSET can be called any time after TELCCB has been called and before the connection is closed or aborted.

Description

The CCB field you requested is set to the value specified in the argument *value*.

6.3.9 TELSND Routine

The TELSND library routine is used to send data to the remote host.

Format

ret-status = TELSND(*cid,buffer,byte-count*)

Arguments

cid

The connection ID (starting address of the CCB).

buffer

The starting address of your buffer containing data to be sent to the remote host.

byte-count

The length of the buffer in bytes. The *byte-count* argument is the address of a word containing the number of bytes in the data buffer.

Description

This routine sends data in your buffer by copying the data into an internal transmit buffer. This routine returns when all data is copied into the internal transmit buffer.

Usually data is copied to the internal buffer immediately, and control is returned to your application program. However, if the internal transmit buffer is full, TELSND waits until there is room in the internal buffer before copying the data and then returns. Buffer space will become available as data is delivered to the remote host.

IAC (ASCII 255) characters in your buffer will be doubled and carriage returns not followed by a line feed will have a NULL inserted following the carriage return as dictated by the TELNET protocol.

TELSND can be called any time after TELOPN has completed successfully, and before the connection is aborted or closed.

6.4 User Command Processing Routine

The user command processing routine is a routine that you write to process received TELNET commands. TELNET commands include option-negotiating commands (such as WILL, WONT, DO, and DONT) and control functions (such as AYT, EL, and BREAK).

Set your command processing routine via a parameter in the TELCCB routine or with the TELSET routine. Your command processing routine is called when TELNET commands are received over the network. Only one TELNET command is passed at a time to your command processing routine. To send a reply, you must use the TELCMD routine to send a TELNET command to the remote host.

The only time your command processing routine will be called is while receiving data with the TELRCV routine. The TELRCV routine calls your processing routine so that you may process TELNET commands at the correct times.

For example, the following data stream will be returned to you in three calls to TELRCV:

$$\underbrace{\mathrm{DDDDDD}}_{1}\underbrace{\mathrm{CCCCDDDD}}_{2}\underbrace{\mathrm{CCDDDDDDDDDD}}_{3}$$

Where "D" represents data bytes and "C" represents TELNET commands. Your command processing routine is called four times at the beginning of call 2 and two times at the beginning of call 3.

This ensures that any change in the way data following a TELNET command is to be interpreted can go into effect before the data is processed. Also, if data needs to be included in the data stream in place of the TELNET command, it can be inserted before the data buffer returned by the TELRCV routine.

Your command processing routine is called with the following format:

Format

Your-Routine(*cid*,*buffer*,*byte-count*);

Arguments

cid

Connection ID (starting CCB address) of the connection on which the command was received.

buffer

Starting address of an internal buffer containing the TELNET command.

byte-count

The length of the TELNET command passed to you in *buffer*.

One complete command at a time is passed to your command processing routine. Commands and options are passed with the leading IAC character removed from them. For example, the command IAC *value* is passed as *value*; IAC DO ECHO is passed as DO ECHO. Subnegotiations are passed as SB *subnegotiation*, where doubled IAC bytes within the subnegotiation are compressed into a single IAC byte.

Your command processing routine must NEVER call the TELRCV routine.

See the *Options and Control Functions* appendix for information on how the command processing routine might handle TELNET options and control functions.

For a complete description of the TELNET protocol, refer to the RFCs listed in the *References* section of the *General Information* chapter.

6.5 Status Codes

The following status codes can be returned by the TELNET-RSX library routines:

Symbol	Value	Description
IS.SUC	1	Success.
SC.CNO	-17	Connection is not open.
SC.CNC	-18	Connection is not closed.
SC.CID	-21	Connection is deleted.
SC.BNP	-50	Bad parameter or number of parameters.
SC.CNI	-60	CCB not initialized.
SC.CLS	-70	Close sent - Local end is closed.
SC.CLR	-71	Close received - Remote end is closed.
SC.TCL	-72	Received invalid TELNET command - too long.

Status code SC.TCL is returned by TELRCV when a TELNET subnegotiation overflows the internal receive buffer. In this case, the complete subnegotiation is discarded. All data and/or commands returned with this error code are valid. This status code is for informational purposes only.

All negative status codes (SC.xxx) except SC.TCL, indicate that the requested operation failed.

Status code SC.CID indicates that the connection is deleted. You must now call TELABO to abort the connection and “free up” the CCB.

Error codes from -1001 to -1127 are RSX I/O errors -1 to -127, respectively.

Error codes from -2001 to -2127 are RSX directive status (DSW) errors -1 to -127, respectively.

6.6 Linking with the Library

The TELNET-RSX library routines are provided in the TELLIB.OLB file. TELLIB.OLB is an object library file, which consists of the modules required to use the TELNET-RSX library routines.

To link your application program with the TELNET-RSX library routines, include the LB:[1,1]TELLIB.OLB/LB library specification in the list of object files used by the program.

A
Options and Control Functions

This appendix describes how User- and Server-TELNET handle TELNET protocol options and control functions.

This appendix is for people who thoroughly understand the TELNET protocols. Refer to the documents listed in the *General Information* chapter, *References* section for more information on these protocols.

A.1 TELNET Options Supported

User- and Server-TELNET support these options:

- ECHO, described in RFC-857
- SUPPRESS-GO-AHEAD, described in RFC-858
- BINARY, described in RFC-856

TELNET-RSX uses the TELNET protocol commands WILL, WONT, DO, and DONT to negotiate options between the user and server.

The following sections describe how each option is implemented.

A.1.1 ECHO Option

The ECHO option enables or disables the echoing of data transmitted over the network. You can configure each side of a connection independently. The initial setting is no echoing.

User-TELNET supports echoing for characters it sends over the network, but not for characters it receives.

Server-TELNET supports echoing for characters it receives over the network. It does not permit User-TELNET to echo characters that User-TELNET receives over the network.

The echo characteristic of the RSX terminal device driver is set or cleared on the terminal belonging to the user or the interactive process, depending on the echo requirements.

Both User- and Server-TELNET try to negotiate for Server-TELNET to echo the characters it receives from User-TELNET.

A.1.2 SUPPRESS-GO-AHEAD Option

The SUPPRESS-GO-AHEAD option enables or disables the sending of the TELNET go-ahead (GA) control function. You can configure each side of a connection independently. The initial setting is to enable suppress go-ahead.

Both User- and Server-TELNET support the negotiation of this option. Both the user and server ignore GA if they receive it and never transmit it, regardless of how this option is set.

Both the user and server try to negotiate for GA to be suppressed.

A.1.3 TRANSMIT-BINARY Option

The TRANSMIT-BINARY option enables or disables the full 8-bit ASCII character set. You can configure each side of a connection independently. Initially, the TELNET protocol disables this option.

User- and Server-TELNET always allow the full 8-bit ASCII character set to be transmitted and received over the network, regardless of how this option is set.

When User-TELNET enables this option, it enables the 8-bit mode on the terminal device. When Server-TELNET enables this option, the 8-bit mode is not affected.

A.2 TELNET Control Functions

The TELNET protocol defines several control functions. These functions are used to interrupt a process, abort output, and erase a character or line.

User-TELNET ignores these functions if it receives them.

Server-TELNET never sends these functions. Server-TELNET replaces the functions it receives in the input stream with the RSX character sequence shown in the following table. One function is an exception: the are you there (AYT) function, which places a BELL character (ASCII 7) in the output stream.

TELNET-RSX supports the TELNET control functions listed in the following table:

TELNET Control Function	**RSX Character Sequence**
Interrupt Process (IP)	^C
Abort Output (AO)	^O
Erase Character (EC)	DEL
Erase Line (EL)	^U

All other TELNET control functions are ignored.

B

Implementation Notes

This appendix describes the TELNET-RSX implementation restrictions due to the RSX environment.

This appendix does not describe the TELNET protocol. Refer to the documents listed in the *References* section in the *General Information* chapter for complete protocol descriptions.

B.1 Special Characters

The ^O, ^Q, ^S, and ^X characters have special meaning to RSX and cannot be intercepted by the User-TELNET utility. These characters are not passed to the remote host when typed at the keyboard. ^O, ^Q, and ^S retain their normal functions.

B.2 Memory Usage

This table shows the approximate amount of memory each TELNET-RSX software module requires. These sizes are rough estimates.

Module	**Memory Requirements**	
TELNET User Utility	20K bytes	(using FCSRES)
INTELD (server)	4K bytes	+ 1280 bytes per terminal
IN Device Driver	10K bytes	
INSACP Task	10K bytes	
XQ Device Driver	4K bytes	
XU Device Driver	6K bytes	
TELNET-RSX Library	4K bytes	

C
Release Notes

This appendix summarizes the enhancements and changes for each version of TELNET-RSX. If you have used earlier versions of TELNET-RSX, read this appendix to determine whether any changes affect you.

Changes affecting the TCP-RSX software product are documented in the *TCPware for RSX Reference Manual, Release Notes* appendix.

C.1 Version 2.2-5

- No changes were made to the TELNET-RSX components in this maintenance release.

C.2 Version 2.2-4

This section describes enhancements and changes for TELNET-RSX Version 2.2-4.

- A problem that occurred when configuring more than 22 TELNET sessions with the TELNET-RSX server was resolved.

C.3 Version 2.2

This section describes enhancements and changes for TELNET-RSX Version 2.2.

- TELNET-RSX supports a server. Users on remote systems can access RSX systems running the TELNET-RSX server. The remote computer can use any hardware or operating system that implements TCP/IP and the TELNET protocol. You can configure the server to support as many remote connections as your resources allow.

  The TELNET-RSX server runs on the RSX-11M-PLUS (V3.0 or later) and MicroRSX operating systems.

  Refer to the *General Information* chapter for more information on Server-TELNET.

- TELNET-RSX contains additional minor improvements and corrections.

C.4 Version 2.1

This is the first release of TELNET-RSX. No previous version exists.

Index