PowerNet TCP/IP

Reference Manual

Connect number CNT5000-5403
Version 5.1 February 1998

Important - Read carefully before opening sealed carton(s). by opening the sealed carton(s) containing enclosed software and/or hardware you indicate your acceptance of the following Connect Inc. license and warranty (Agreement).

This is a legal agreement between you (either an individual or an entity) and Connect Inc. by opening this sealed carton you are agreeing to be bound by the terms of the agreement. if you do not agree to the terms of this agreement, promptly return the unopened carton(s) and the accompanying items (including printed materials and binders or other carton(s)) to the entity from which you obtained them for a full refund.

Connect Inc. License and Warranty

1. GRANT OF LICENSE. This license permits you to use one copy of the enclosed Connect/RF software program (the "SOFTWARE") on a single computer. The SOFTWARE is in "use" on a computer when it is loaded into temporary memory (i.e., RAM) or installed into permanent memory (e.g., hard disk, CD-ROM, or other storage device) of that computer. However, installation on a network server for the sole purpose of internal distribution shall not constitute "use" for which a separate license is required, provided you have a licensed copy of the SOFTWARE for each computer that can access the Software over that network.

2. COPYRIGHT. The SOFTWARE and all copyright, trade secrets and other right, title and interest therein is owned by Connect Inc., or its licensors, or suppliers and is protected by United States copyright laws and international treaty provisions. Therefore, you must treat the SOFTWARE like any other copyrighted material (e.g., a book or musical recording) except that you may either (a) make one copy of the SOFTWARE solely for backup or archival purposes, or (b) transfer the SOFTWARE to a single hard disk provided you kept the original solely for backup or archival purposes.

3. OTHER RESTRICTIONS. You may not sublicense, rent or lease any portion of the SOFTWARE, but you may transfer the SOFTWARE and accompanying written materials on a permanent basis provided you retain no copies and the recipient agrees to the terms of this Agreement. You may not reverse engineer, decompile, disassemble, or modify, translate, make any attempt to discover the source code of the SOFTWARE or create derivative works from the SOFTWARE. You may not copy the documentation which accompanies the SOFTWARE. If the SOFTWARE is an update or has been updated, any transfer must include the most recent update and all prior versions.

4. DUAL-MEDIA SOFTWARE. You may receive the SOFTWARE in more than one medium. Regardless of the type or size of medium you receive, you may use only the medium appropriate for your single designated computer or network server. You may not use the other medium on any other computer or computer network, or loan, sublicense, rent, lease, or transfer it to another user except as part of a permanent transfer (as provided above) or other use expressly permitted by this License.

5. EXPORT RESTRICTIONS. You agree that neither you nor your customers intends to or will, directly or indirectly, export or transmit (a) the SOFTWARE or related documentation and technical data or (b) your software product, or process, or service that is the direct product of the SOFTWARE, to any country to which such export or transmission is restricted by any applicable U.S. regulation or statute, without the prior written consent, if required, of the Bureau of Export Administration of the U.S. Department of Commerce, or such other governmental entity as may have jurisdiction over such export or transmission.

6. LIMITED WARRANTY. Connect warrants that (a) the media on which the Software is distributed will perform substantially in accordance with the accompanying written materials for a period of (30) days from the date of receipt of the SOFTWARE, and (b) any hardware accompanying the SOFTWARE will be free from defects in materials and workmanship under normal use and service for a period of one (1) year from the date of receipt. Any implied warranties on the SOFTWARE and hardware are limited to thirty (30) days and one (1) year, respectively. Some states/jurisdictions do not allow limitations on duration of an implied warranty, so the above limitation may not apply to you. Connect Incorporated does not warrant that the SOFTWARE will meet your requirements or that operation of the SOFTWARE will be uninterrupted or that the SOFTWARE will be error-free.

7. CUSTOMER REMEDIES. Connect Inc., its licensors' and suppliers' entire liability and your exclusive and sole remedy in the event of a breach of the Limited Warranty will be, at Connect's option, either (a) refund of the price paid for the SOFTWARE and/or the hardware, or (b) repair or replacement of the SOFTWARE or hardware that does not meet Connect's Limited Warranty and which is returned to Connect within the warranty period with a copy of your purchase order/receipt. This Limited Warranty is void if failure of the SOFTWARE or hardware has resulted from accident, abuse, or misapplication. Any replacement SOFTWARE or hardware will be warranted for the remainder of the original warranty period or thirty (30) days, whichever is longer. Outside the United States, neither these remedies nor any product support services offered by Connect are available without proof of purchase from an authorized non-U.S. source.

8. No other warranties. the above warranty is exclusive and in lieu of all other warranties, whether express or implied, including, but not limited to, implied warranties of merchantability, fitness for a particular purpose and non-infringement, this limited warranty gives you specific legal rights. you may have others which vary from state/jurisdiction to state/jurisdiction.

9. Disclaimer of Damages. Regardless of whether any remedy set forth herein fails of its essential purpose, in no event shall Connect incorporated or its suppliers or licensors be liable for any special, consequential, indirect or similar damages whatsoever (including without limitation, damages for any loss of business profits, business interruption, loss of business information or data or any other pecuniary loss) arising out of the use or inability to use the software and/or hardware, even if Connect Inc., has been advised of the possibility of such damages. because some states/jurisdictions do not allow the exclusion or limitation of liability for consequential or incidental damages, the above limitation may not apply to you.

10. U.S. GOVERNMENT RESTRICTED RIGHTS. The SOFTWARE and documentation are provided with RESTRICTED RIGHTS. Use, duplication, or disclosure by the Government is subject to restrictions as set forth in subparagraph (c)(1)(ii) of the Rights in Technical Data and Computer Software clause at DFARS 252.227-7013 or subparagraphs (c)(1) and (2) of the Commercial Computer Software-Restricted Rights clause at 48 CFR 52.227-19, as applicable. Manufacturer is Connect Inc., 5400 Patton Dr., Suite 300, Lisle, IL 60532.

GENERAL:

If you acquired this product in the United States, this Agreement is governed by the laws of the State of Illinois. This Agreement may only be modified by a license addendum which accompanies this license or by a written document which has been signed both by you and Connect Inc.

If you have any questions concerning this Agreement, or if you desire to contact Connect for any reason, please write; Connect Inc., 5400 Patton Dr., Suite 300, Lisle, IL 60532.

Production This manual was written, edited, and produced by
Connect, Inc.
5400 Patton Dr., Suite 300
Lisle, IL 60532

Printed in the U.S.A.

Connect number CNT5000-5403
Version 5.0 April 1997

Contents

1 TCP/IP Connectivity Overview

2 Basics

3 TCP/IP Connection Setup

4 Configuration

Appendix Client Streaming API

This page intentionally left blank.

1 TCP/IP Connectivity Overview

Introduction

PowerNet provides connectivity between Spectrum One or other 900MHz LANs and a host through a network running TCP/IP, when equipped with a *TCP/IP Connectivity Package*. There are six types of *TCP/IP Connectivity Packages*, for varying environments. Two *Client Streaming Kits*, TCP/IP for Ethernet and TCP/IP for Token Ring and four *Terminal Emulation Kits*, TCP/IP for Token Ring, TCP/IP for Ethernet, TN3270 for Ethernet, and TN3270 for Token Ring.

This *PowerNet TCP/IP Reference Manual* describes the procedures necessary to install and support connections to a TCP/IP network using this software.

Features

TCP/IP Connectivity Packages offers the following:

- **Easy Form-based Setup** - Forms are used to enter information for system setup. This process is completed from within the program. There are no files to edit or command line operations to be performed.

- **Token Ring & Ethernet** - Connectivity packages are available for both Token Ring and Ethernet networks. Thicknet, thinnet, and 10 Base-T Ethernet as well as token ring 4 and 16, are supported.

- **Supports Standard Functions** - *TCP/IP Connectivity Packages* conform to industry standards. There are no proprietary modifications. TELNET, rlogin, FTP, RCP, and SNMP are supported.

Caution While TCP/IP services such as TELNET and RLOGIN are available as part of the standard TCP/IP PowerNet package, care must be used when accessing PowerNet services such as TCP/IP Shutdown. This service, for example, stops all TCP/IP activity including any RLOGIN or TELNET sessions that may be active. The only way to restart these services would be from the local console or via modem access to the PowerNet server. In all other cases, the PowerNet services can be accessed via these methods.

- **Client Streaming & Terminal Emulation** - A transaction processing interface is available with client streaming packages. This interface gives a faster, more efficient, battery-saving environment. Standard terminal emulation, which makes use of Enablers, is also available.

- **Logs** - Logs help tune the system during setup and are also used for problem resolution. Log levels may be set to collect various amounts of information, or none, during normal operation.

- **Easy Terminal Menu Modification** - Add hosts/applications to the RF terminal *Host List Menu* easily, and without using an editor. Use the PowerNet Host List option and Handlers for these additions and/or modifications to the menu.

- **Screen Formatter Compatible** - The *Screen Formatter Package* allows customization of full-size host screens to smaller RF terminal screens, for TN3270 as well as other emulations. Although it is purchased separately, it is fully integrated with these packages in both operation and function.

Menus

Modifications to the PowerNet server base menus, shown in the *PowerNet System Manual*, include *TCP Network* which is added to *the Main Menu* and two additions to the *Operations Menu*, following the *Startup RF Network* selection. These only appear in client streaming but not in terminal emulation software. The additions are *TCP Startup and TCP Shutdown*.

Documentation

A brief review of terms appears in Chapter 2. Instructions for setting up PowerNet network connection parameters including addresses and netmasks, follows. After setup, checking connections and status is performed. *Chapter 4* provides instructions for modification of the *Host List* file which involves *Handlers*. Information regarding installed network hardware is needed while reading the manual and setting up the system. The Network Administrator should provide this information. In addition, readers may find a background in TCP and IP protocols to be helpful.

Conventions

This document uses the same typographical conventions as the *PowerNet System Manual*.

Interface

Please see the *PowerNet System Manual* for information about the screen formats and keys used with this software.

Product Information

The *PowerNet TCP/IP Connectivity Packages* for the PowerNet and associated packages are listed below with their products numbers.

Product #	Description
CNT5000-3103	TCP/IP Ethernet Client Streaming Package
CNT5000-3104	TCP/IP Token Ring Client Streaming Package
CNT5000-3202	TCP/IP Ethernet Terminal Emulation Package
CNT5000-3204	TCP/IP Token Ring Terminal Emulation Package
CNT5000-3111	TN3270 Ethernet Terminal Emulation Package
CNT5000-3112	TN3270 Token Ring Terminal Emulation Package
CNT5000-3502	VTerm Terminal Emulation Software Screen Formatter Software Package

Note Technical Information for the TN3270 packages, can also be found in the *PowerNet 3270 Reference Manual*.

This page intentionally left blank.

2 Basics

Introduction

This chapter describes some of the concepts and terminology that are used in this manual.

Client Streaming vs. Terminal Emulation

Two types of software packages are available for TCP/IP connectivity, client streaming (CS) and terminal emulation (TE). These software packages differ in the mode of communication used and the environment that is implemented. Selection of a package depends in part on what the host application and site require.

Terminal Emulation under the PowerNet emulation architecture reduces the high overhead usually associated with this type of connectivity. When terminal emulation is being implemented, the RF terminals appear to the host as terminals off a terminal server, which is the PowerNet server. All emulation kits use the Terminal Access Program (TAP), supplied with the PowerNet server, to run on the remote terminal devices.

The combination of the TAP program and PowerNet translation and control software convert the various terminal datastreams to a highly efficient common format designed specifically for wireless handheld terminals. This common client architecture makes possible features such as hot-key between similar or dissimilar host applications and emulations, as well as screen formatting, keyboard mapping, and a variety of other features. TAP is supplied with all systems in two versions: one for Spectrum One or other spread spectrum terminals and another for access point addressable terminals. Functionally, TAP is identical for both wireless technologies.

Client Streaming is suited for a high volume of traffic while providing maximum performance. However, implementation of client streaming applications requires specific applications to be created for both the host and remote terminal devices. An API is available for custom application development or a specially designed STEP enabler may be used. During system operation, with a client streaming package, the PowerNet server appears to the host as just another node on the network. RF terminals appear as transaction terminals.

TCP/IP

Using TCP/IP for connectivity is one of the most versatile methods available today. Connections to hosts on the same local area network (LAN), on wide area networks (WANs), on foreign (different network operating systems (NOSs) networks, or to minicomputers and mainframes are possible.

The TCP/IP protocol suite implemented on the PowerNet server utilizes the Transmission Control Protocol (TCP), and Internet Protocol (IP), and also includes the Point-to-Point Protocol (PPP), Serial Line Internet Protocol (SLIP), Address Resolution Protocol (ARP), Simple Network Management Protocol (SNMP), and others. TCP/IP commands and protocols that provide end-user networking capabilities have also been implemented. These include *ftp*, *rcmd*, *rcp*, *rlogin*, *ruptime*, *rwho*, and *telnet*. Note that these are not explained here, as they are beyond the scope of this manual.

In most installations, a TCP/IP network is already in place, and the PowerNet server must be added. The TCP/IP Network Administrator needs to be available in order to provide the details of the network and addresses of the host(s) and other nodes that are needed. PowerNet also needs parameters defined by the system administrator, e.g., address(es) and netmask(s).

Addresses

The addressing scheme used is defined by the Internet Protocol (IP). The unique address assigned to the PowerNet server distinguishes it from all other machines on the network to which it is attached. The IP address is obtained from the TCP/IP network administrator. It is a 32-bit binary number and is displayed as four octets expressed in decimal, separated by periods. This address identifies the network and the individual machine which is referred to as a node or host (node/host).

Three classes of TCP/IP networks (and their corresponding addresses) are supported by the software being used. These are class A,B, and C which are defined by the number of hosts/nodes a network needs. The following table defines the IP Address Classes.

Class	Hosts Per Network	Valid Address Ranges	Address Usage
A	1677216	1.0.0.1 - 126.255.255.254	aaa.nnn.nnn.nnn Network Address = First Octet Node Name = Last 3 Octets

B	65534	128.0.0.1 - 191.255.255.254	aaa.aaa.nnn.nnn Network Address = First 2 octets Node Name = Last 2 Octets
C	254	192.0.0.1 - 222.255.255.254	aaa.aaa.aaa.nnn Network Address = First 3 octets Node Name = Last Octet
Reserved		224.0.0.0 - 255.255.255.254	

In most installations, a TCP/IP network is already in place. Therefore, the network part of the PowerNet address, is the same as other machines on that particular network, while the node/host part of the IP address is unique. All of the PowerNet address information should be provided by the TCP/IP Network Administrator.

Netmask

When configuring the PowerNet server for TCP/IP, a netmask is required on the *PowerNet TCP/IP Setup Form*. A netmask masks out the network address from the IP address, leaving only the node address. Each netmask consists of binary ones (decimal 255) to mask the network ID and binary zeros (decimal 0) to retain the node/host ID. An example of a class C netmask would be 255.255.255.0

A Word of Caution

While TCP/IP services such as TELNET and RLOGIN are available as part of the standard TCP/IP PowerNet package, care must be used when accessing PowerNet services such as TCP/IP Shutdown. This service, for example, stops all TCP/IP activity including any RLOGIN or TELNET sessions that may be active. The only way to restart these services would be from the local console or via modem access to the PowerNet server. In all other cases, the PowerNet services can be accessed via these methods.

This page intentionally left blank.

3 TCP/IP Connection Setup

Introduction

This chapter describes the procedures required to set up a communication link between the PowerNet and TCP/IP host(s)/network(s)/Internet. Depending upon whether a Client Streaming or Terminal Emulation package is being implemented, certain steps are performed, or left out. These are noted in the text.

Caution While TCP/IP services such as TELNET and RLOGIN are available as part of the standard TCP/IP PowerNet package, care must be used when accessing PowerNet services such as TCP/IP Shutdown. This service, for example, stops all TCP/IP activity including any RLOGIN or TELNET sessions that may be active. The only way to restart these services would be from the local console or via modem access to the . In all other cases, the PowerNet services can be accessed via these methods.

Since Ethernet and Token Ring connectivity setup is similar, one set of instructions is given. Any differences are documented. Menus and forms are almost identical, with differences indicating either Ethernet or Token Ring, and examples of both are included.

If Token Ring connectivity is being implemented, one additional step must be taken. This is to determine whether the Token Ring Network is operating at 4 or 16 Mbps. If it is 16, no additional work must be performed, other than the steps listed in the summary below. If the network is operating at 4 Mbps, a DIP switch on the token ring NIC must be changed. Please see *Token Ring NIC* on Page 3-3 for instructions.

The following is a summary of the link setup process:

- **Shut Down Host Link** - Terminate any activity between the host and PowerNet server for client streaming packages. This option is not included, nor is necessary, for emulation packages.

- **Set Node Parameters** - Complete the setup form with parameters obtained from the TCP/IP Network Administrator.

- **Set Up Physical Connection** - Connect all hardware, connectors, cables, etc., as required, for the link from the PowerNet server to the TCP/IP network/host.

- **Start Up Host Link** - Activate the connection between the TCP/IP network/host and the PowerNet server for client streaming packages. This option is not included, for terminal emulation packages.

- **Define Environment** - Complete the address forms for other nodes on the network.

- **Port Setup** - Define corresponding host and PowerNet ports (sockets) for CS packages only.

- **Verify Link** - Display status screens and perform ping tests for checking the communication path between the PowerNet server and the network node/host.

Hardware

Equipment and cabling *external* to the Ethernet or Token Ring NIC, is **not** provided. The necessary equipment and its configuration is specific to the network/host and site requirements. Therefore, the hardware connections must be made on an individual basis. Only general information is given here. The PowerNet server may have one or two adapters and can be connected to a maximum of two individual TCP/IP networks/hosts.

Ethernet NIC

The Ethernet NIC can be connected to the network using either thin or thick Ethernet cable or unshielded twisted pair wire (UTP). The card provides an RJ-45 plug for UTP, a BNC connector for thin coaxial cable, and an AIU female connector for thick coaxial cable. All cables are customer-provided. Pinouts for the AIU female connector are as follows:

AUI Connector

Pin	Function
1	Control In Shield
2	Control In A
3	Data Out A
4	Data In Shield
5	Data In A
6	DC Power Common
7	No Connection
8	No Connection
9	Control In B
10	Data Out B
11	Data Out Shield
12	Data In B
13	DC Power +
14	Power Shield
15	No Connection

Token Ring NIC

Each TCP/IP-Token Ring PowerNet server comes standard with a NIC (Network Interface Card) that provides one DB-9 female connector (STP) and a RJ-45 connector (UTP). The cables that connect from the PowerNet server to the Multi-station Access Unit (MAU) for the Token Ring are customer provided. To change the operating data rate for the card or to change the connection type, open the *Local Setup* option from the *TCP/IP* menu at the *PowerNet Main Menu*. The setup screen as shown in figure 3-1 appears.

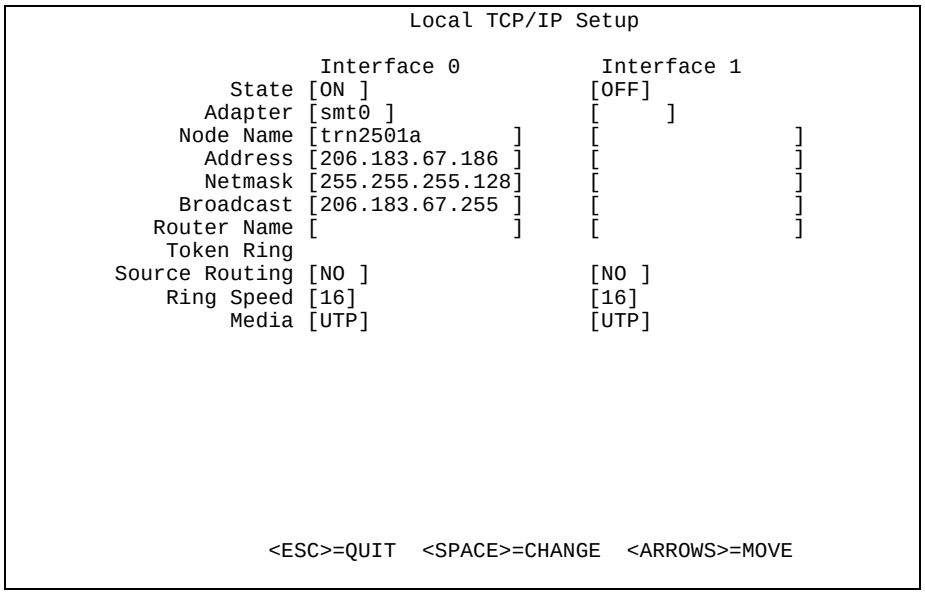


Figure 3-1 TCP/IP Local Setup

Use the <arrow> keys to position the cursor to the Ring Speed field and use the <space> key to toggle the field value to the proper ring speed (4Mbps or 16Mbps). The Media field allows selection of the connector on the card to access the network. Toggle this field using the <space> key to select the proper field value (STP or UTP). Details on the remaining fields for this screen are discussed later in this chapter.

TCP/IP Network Menu

The *TCP/IP Network Menu* option provides access to setup, configuration, test and status options that are summarized below. Select *TCP/IP* from the *Main Menu* and the menu shown in Figure 3-2 is displayed.

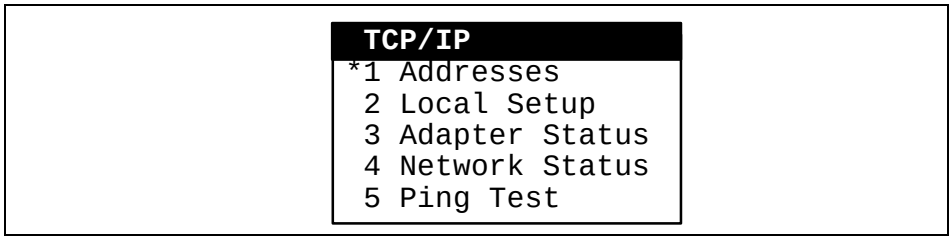


Figure 3-2 TCP Network Menu

Menu Options

Addresses Addresses of the PowerNet server and other nodes on the TCP/IP network are entered on this form for reference and use with other features of PowerNet such as TN3270.

Local Setup This option displays a form on which PowerNet address information is specified. The TCP/IP Network Administrator must supply the information that is entered on this form.

Adapter Status Check the status of adapter cards in the PowerNet server with this option.

Network Status Use this option to check the current status of the TCP/IP network.

Ping Test Connections between the PowerNet server and nodes listed on the Node Addressing Form can be tested with this option. Packets are sent between the two nodes.

Network Management - TCP Shutdown

Caution Be forewarned that shutting down TCP/IP services while accessing the PowerNet server via TELNET or RLOGIN immediately disconnects the session from the PowerNet server. The only way to reliably stop and restart the TCP/IP services remotely is through the internal modem, service port, or at the PowerNet's local console.

Before setting up the connection, select the *TCP/IP Shutdown* option from the *Operations Menu*, and then press <Enter>. This terminates any link to the TCP/IP network. (Note that if you are logged in via TCP/IP (using *rlogin* or *telnet*) a disconnect occurs.) A prompt for a confirmation of the shutdown action appears on the screen. An example is shown below.

```
Are you sure you want to shut down TCP/IP (y/n)?
```

Press <**y**> followed by <Enter> and the system responds with

```
Shutting Down TCP/IP
TCP Shutdown...
slink exiting: SIGTERM
Stopping TCP/IP Daemons ...
......
TCP Shutdown Complete.
             Press <Enter> to continue
```

Press <Enter> and the *Operations Menu* reappears. After all possible activity is terminated, the setup parameters for the link to the TCP/IP network can be entered.

TCP/IP Addresses Setup

Select *Addresses* from the *TCP/IP Network Menu* to display the *TCP/IP Network Addresses Setup Form* shown in Figures 3-3. Connection information, supplied by the TCP/IP Network Administrator, goes here.

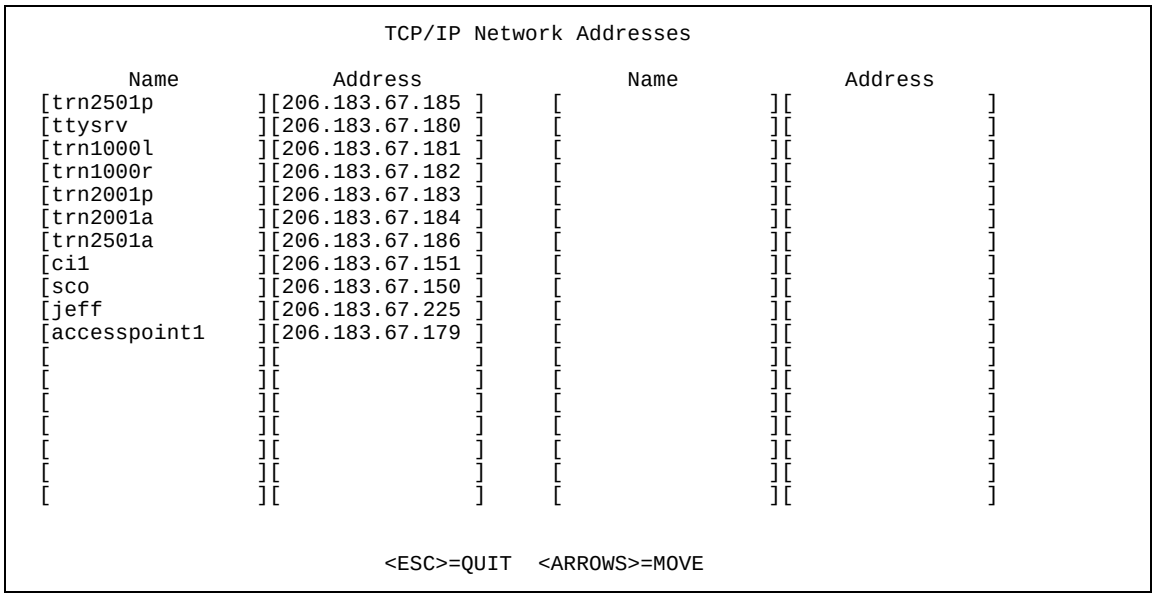


Figure 3-3 TCP/IP Addresses Setup Form

It is important while working at this screen to remember not only to input the addresses of the TCP/IP devices that the PowerNet server attaches to but also the name and the address of the PowerNet server itself, as this entry is used by the PowerNet server to identify itself when performing the Local Setup functions.

Once the PowerNet server has been set up as a TCP/IP node(s) on the network, other nodes and routers with which it can communicate are listed on the *TCP/IP Node Address Form*.

TCP/IP Addresses Form - Fields

This Addresses Setup form is for communication information. Names and addresses of host nodes, with which the PowerNet/RF terminals establish sessions, should be listed. In addition, other nodes can be added for testing purposes. There are fields for 36 nodes.

Name [Text Field] Enter the name of a node on the network/Internet which is provided by the TCP/IP Network Administrator.

Address [Text Field] Enter the unique IP address for each node.

Note Enter the names and addresses on this form of the PowerNet adapter. If two were configured, enter both.

TCP/IP Local Setup

Select *Local Setup* from the *TCP/IP Network Menu* to display the *TCP/IP Local Setup Form* shown in Figures 3-4. Connection information, supplied by the TCP/IP Network Administrator, is entered here.

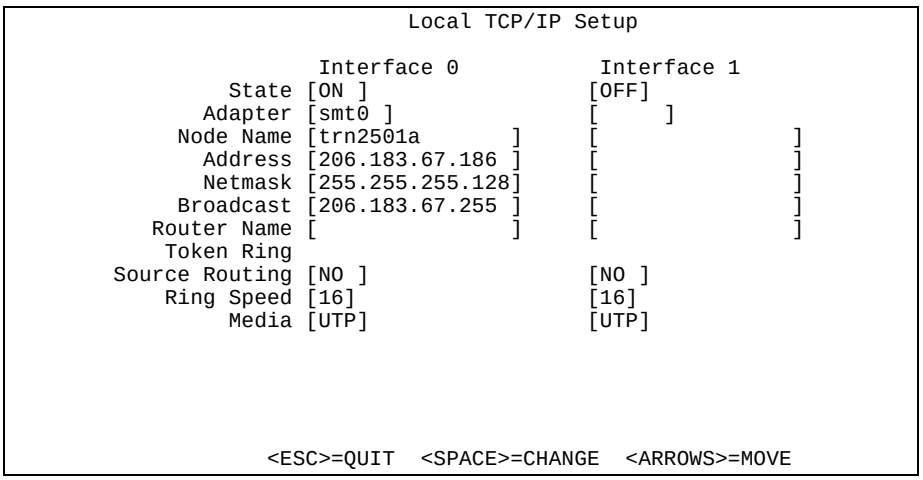


Figure 3-4 Local TCP/IP Setup Form

Local TCP/IP Setup Form - Fields

The fields follow, with the type of field, toggle, text, or display listed.

State [Toggle Field] The link may be set to either ON or OFF.

Adapter [Toggle Field] Toggles all possible selections and is site specific. Selections are wdn, sme, smpw, e3G, tok, and smt. The available values for this field are system dependent and only display the values of the adapters installed in the PowerNet server. The adapter name is immediately followed by a 0 or 1 to indicate which adapter card is selected, as PowerNet can be configured with up to 2 TCP/IP adapters (example, sme0).The type of media is factory set, and the selections that appear should be correct. If not, select the type of adapter being configured.

Node Name [Toggle Field] This field is toggled with values that were set in the Addresses Form, using the TCP/IP pseudo-name as the value. Select the unique name provided by the TCP/IP Network Administrator for each adapter being configured.

Address	[Fill Field]	This field is filled with the IP address of the pseudo-name selected in the node name field above. This field is provided for informational purposes only and cannot be changed here.
Netmask	[Text Field]	Enter the netmask provided by the TCP/IP Network Administrator for the adapter being configured. By default, if these fields are blank, they are automatically considered for a class C addresss. See Chapter 2 for more information.
Broadcast	[Text Field]	Enter the broadcast address of the network segment to which the PowerNet server is attached.
Router Name	[Toggle Field]	This is an optional field in which the name of a server, that maintains the current router tables, is entered. The TCP/IP Network Administrator provides this information. The actual router IP address and pseudo-name is set in the TCP/IP Addresses form.
Source Routing	[Toggle Field]	This field is used for smt*n* Token Ring TCP/IP configurations only. This field enables (on) or disables (off) source routing. Contact the network administrator to find out whether source routing is used for their particular network.
Ring Speed	[Toggle Field]	This field is used for smt*n* Token Ring TCP/IP configurations only. This field specified the netowrk ring speed at which the adapter should operate - either 4Mbps or 16Mbps. Contact the network administrator in order to determine the speed used for their particular network.
Media	[Toggle Field]	This field is used for smt*n* Token Ring TCP/IP configurations only. This field selects the connector type the adapter will use to connect to the network. Values for this field are STP (for the DB9 connector) or UTP (for the RJ45 connector).

When the form is complete, press <Esc>. If changes were made to the form, a confirmation prompt is presented. Press <Y> to save the changes and the *TCP Network Menu* reappears.

Network Management - TCP Startup

After configuration is complete, the network link must be started. Return to the *Operations Menu* and select *TCP/IP Startup*, then press <Enter>. The following is shown on-screen.

```
Activating TCP-IP network
Starting TCP/IP: cpd slink ldsocket

CARD=sme0 ADDR=2-6.183.67.185 MASK=255.255.255.128
BROADCAST=205.183.67.255

CARD=smt0 ADDR=2-6.183.67.187 MASK=255.255.255.128
BROADCAST=205.183.67.255
strerr syslogd inetd routed

Press ENTER to continue
```

Press <Enter> and the *Operations Menu* reappears.

Adapter Status

Select *Adapter Status* from the *TCP/IP Menu* to check the status of PowerNet's Ethernet and/or token ring adapters. The *Adapter Status* display is shown below. The system gathers this information by directly querying the adapter card.

```
Device       SNPA/MAC address    Factory Address
/dev/sme0    00:00:c0:9e:04:bf   00:00:c0:9e:04:bf
        Multicast address table
                01:a0:f8:f0:f0:02
        Frames:      In      Out  Errs In  Err out Collisions
                   79720    15077        0       10        351
Device       SNPA/MAC address    Factory Address
/dev/smt0    00:00:03:4e:5a:9d   00:00:03:4e:5a:9d
        Multicast address table
        Frames:      In      Out  Errs In  Err out Collisions
                      0        0        0        0          0

Press ENTER to continue
```

Figure 3-5 Adapter Status Display

Adapter Status Display - Fields

Device The UNIX device name of the adapter as known to the PowerNet server.

SNPA/MAC Address The SubNet Point of Attachment/Medium Access Control address.

Factory Address The adapter card hardware address set at the factory.

Frames In The number of incoming frames (packets).

Frames Out The number of outbound frames.

Errors In The number of errors experienced with incoming frames.

Errors Out The number of errors experienced with outbound frames.

Collisions The number of collisions on the network.

Press <Enter> to return to the TCP Network Menu.

Network Status

When *Network Status* is selected from the *TCP Network Menu*, the display shown in Figure 3-6 appears. The information is collected from the TCP/IP drivers at the software level for each network interface card. The fields are explained below.

```
Name   Mtu   Network     Address        Ipkts   Ierrs Opkts   Oerrs
Collis
sme0   1500  206.183.67  trn2501p       80955   0     15260   10     352
lo0    8232  loopback    me             1015    0     1015    0      0

Press ENTER to continue
```

Figure 3-6 Network Status Display

Network Status Display - Fields

Name The name of the network interface.

Mtu Maximum transmission unit (in byte)s. This is the largest size permitted for packets.

Network The network part of the IP address. Or, for each adapter, a loopback path is available and is shown as *lo0* or *lo1*.

Address The node/host part of the IP address which is unique.

Ipkts The number of incoming packets received on this interface.

Ierrs The number of errors for incoming packets received on this interface.

Opkts The number of outbound packets sent on this interface.

Oerrs The number of errors for outbound packets.

Collis The number of collisions that occurred on the network.

Press <Enter> to return to the *TCP Network Menu*.

Ping Test

The initial *Ping Test* screen is shown in Figure 3-7. The *Ping Test* sends test packets to the host/node that is entered on the screen. If the packets are received, the connection is good. If return packets are not received, failure of the connection needs to be investigated further.

```
CURRENT HOSTS FOR PING TEST:

127.1           me loopback localhost
206.183.67.185  trn2501p
206.183.67.180  ttysrv
206.183.67.181  trn1000l
206.183.67.182  trn1000r
206.183.67.183  trn2001p
206.183.67.184  trn2001a
206.183.67.186  trn2501a
206.183.67.179  accesspoint1

Host name: ttysrv
```

Figure 3-7 Ping Test - Host Selection Screen

Enter the pseudo-name of the host that is to be pinged. In Figure 3-8, a ping test is being performed. After the name is typed in, press <Enter> and the following information should appear.

```
Host name: ttysrv
PING ttysrv (206.183.67.180): 56 data bytes
64 bytes from ttysrv (206.183.67.180): icmp_seq=0 ttl=255 time=0 ms
64 bytes from ttysrv (206.183.67.180): icmp_seq=1 ttl=255 time=0 ms
64 bytes from ttysrv (206.183.67.180): icmp_seq=5 ttl=255 time=0 ms
64 bytes from ttysrv (206.183.67.180): icmp_seq=6 ttl=255 time=0 ms
64 bytes from ttysrv (206.183.67.180): icmp_seq=7 ttl=255 time=0 ms
64 bytes from ttysrv (206.183.67.180): icmp_seq=8 ttl=255 time=0 ms
64 bytes from ttysrv (206.183.67.180): icmp_seq=9 ttl=255 time=0 ms

--- ttysrv ping statistics ---
7 packets transmitted, 7 packets received, 0% packet loss
round-trip min/avg/max = 0/0/0 ms

Press ENTER to continue
```

Figure 3-8 Ping Test Results

Press <Del> to stop the test. Note that if the test is unsuccessful (100% packet loss), check the *Device Status*, physical connections, and the addresses that were entered.

Accessing Applications

Once a system is properly configured for connection to the host environment, application *Handlers* can be set up for accessing host applications either in the client streaming environment or the terminal emulation environment. Refer to the appropriate manual(s) included for setting up application handlers.

This page is intentionally left blank.

4 Configuration

Introduction

Once the TCP/IP communication path between the PowerNet and the host is operational, a method for connecting to the host application needs to be configured (host interface). The three methods for interfacing to host applications are Terminal Emulation, Client Streaming, and STEP Enablers. The PowerNet server is configured to support only those connectivities specified when ordered, so all options discussed here may not apply to your specific PowerNet server.

Once an application interface is configured, the *Host List* can be configured. The *Host List* is tailored to contain the application(s)/host(s) that are available for use by RF terminal operators. Each of these entries is assigned a *Handler*, which in turn can be customized.

Application Interface Setup

TCP/IP - Terminal Emulation

Terminal emulation in the TCP/IP environment is one of the most commonly used application interfaces. PowerNet supports VTERM (VT100, VT220, HP700), TN3270, and TN5250 terminal emulation under TCP/IP networks. Details of each application interface for terminal emulation is discussed below.

VTERM

VTERM (VT100, VT220, HP700) terminal emulation interfaces are defined through Handler setups as there are several methods for accessing host applications using these types of interface. Details for VTERM host interface setups are found in the *VTERM Reference Manual*.

TN3270

The TN3270 Terminal Emualtion Interface Setup is found under the *TN3270* menu option from the *PowerNet Main Menu*. Opening the *Setup* option from this menu displays the *TN3270 Setup Screen* as shown in figure 4-1.

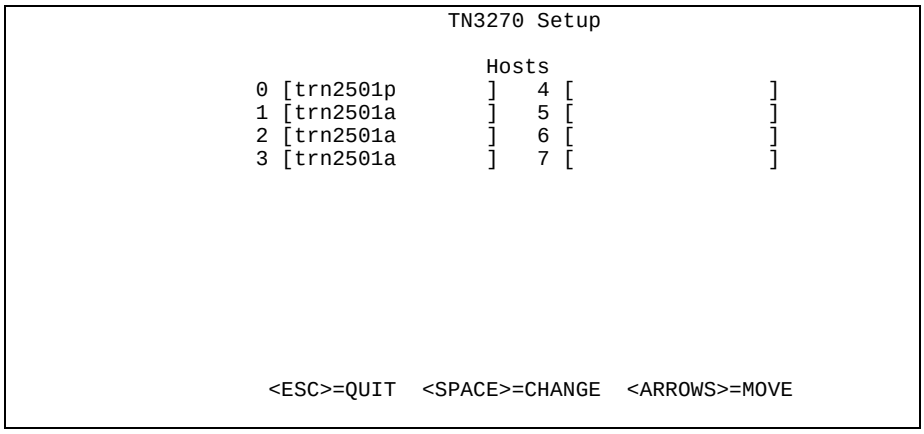


Figure 4-1 TN3270 Setup Screen

At this screen, toggle through the field values until the desired host appears in the field. Up to 8 hosts can be addressed under TN3270. Remember, the values used for this toggle selection list come from the TCP/IP Network Address setup form previously reviewed in Chapter 3. The host number (0-7) is used in the *Handler* setup to tie the terminal emulation application to the TCP/IP host listed here. See the *3270 Reference Manual* for details regarding this host number field.

TN5250

The TN5250 Terminal Emualtion Interface Setup is found under the *TN5250* menu option from the *PowerNet Main Menu*. Opening the *Setup* option from this menu displays the *TN5250 Setup Screen* as shown in figure 4-2.

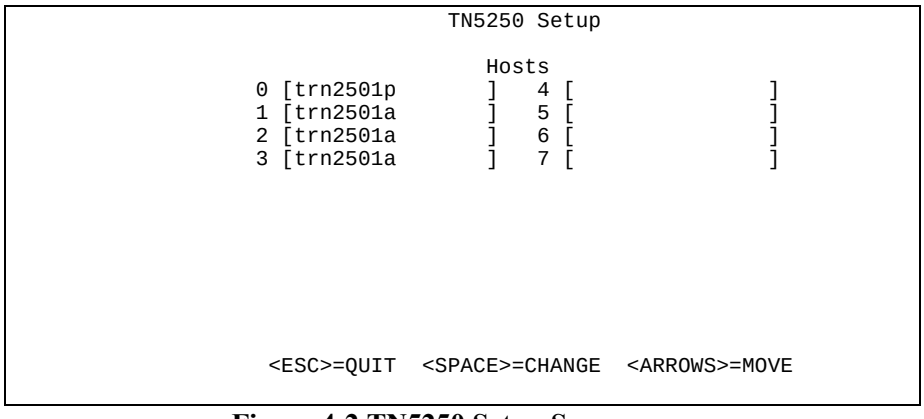


Figure 4-2 TN5250 Setup Screen

At this screen, toggle through the field values until the desired host appears in the field. Up to 8 hosts can be addressed under TN5250. Remember, the values used for this toggle selection list come from the TCP/IP Network Address setup form previously reviewed in Chapter 3. The host number (0-7) is used in the *Handler* setup to tie the terminal emulation application to the TCP/IP host listed here. See the *5250 Reference Manual* for details regarding this host number field.

TCP/IP - Client Streaming

Client Streaming interfaces provide for custom applications to be developed on the host system as well as on the hand held terminal that utilizes PowerNet as a pass-through data server in the TCP/IP environment. To access the TCP/IP Client Streaming Setup function, open the TCP/IP-CS menu selection from the PowerNet Main Menu. The TCP/IP-CS menu is shown in figure 4-3.

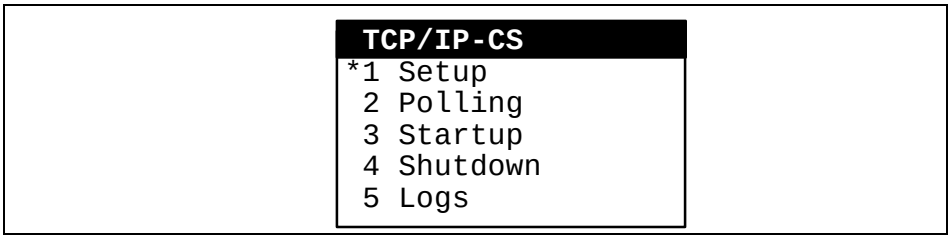


Figure 3-2 TCP/IP-CS Menu

TCP/IP-CS Menu - Setup Option

The TCP/IP-CS setup screen is shown as in figure 4-4.

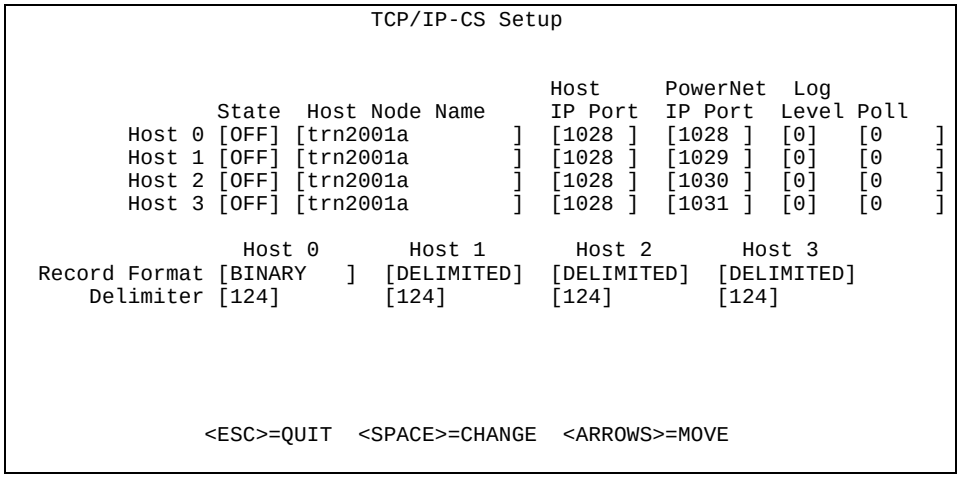


Figure 4-4 TCP/IP Client Streaming Setup Screen

Up to 4 TCP/IP Client Streaming host interfaces can be defined per PowerNet server. The host number (0-4) is used to tie the *Handler* setup to the selected host interface. The fields used at this setup form are as follows:

HOSTn	[Fixed]	This display field shows the number of the defined host.
State	[Toggle Field]	This field is used to define the state of the defined interface. State can be ON (active) or OFF (inactive)
Host Node Name	[Toggle Field]	The values used for this toggle selection list come from the TCP/IP Network Address setup form, previously reviewed in Chapter 3. Select the host name for the client streaming interface using the <space> key to view the selections. Press <enter> to accept.
Host IP Port	[Text Field]	Enter the IP port number for PowerNet to transmit to the host application. Valid values for this field are 180-511, 557-910, 912-1024, 1027-1523, 1525-32768.
PowerNet IP Port	[Text Field]	Enter the IP port number for PowerNet to receive from the host application. Valid values for this field are 180-511, 557-910, 912-1024, 1027-1523, 1525-32768.
Log Level	[Toggle Field]	Sets the log detail level for the TCP host log. Leave this field set to 0 for normal PowerNet operation. Values for this field are 0-9.
Record Format	[Toggle Field]	The record type may be Delimiter, if so each record transferred between PowerNet and host is delimited by the character whose decimal value is defined in the Delimiter field. If record type is Binary, each record contains a 2 byte header that defines the length of the record.
Delimiter	[Text Field]	The ASCII value or the delimiter to be sent if the Record Format field is set to Delimiter.

TCP/IP-CS Menu - Polling Option

The TCP/IP-CS polling option setup screen prompt is shown as in Figure 4-5.

TCP/IP - STEP Enabler

Host List

The *Host List* is constructed and changed with the *Host List Setup Form*. There is no need to use an editor. You must add *Host List* options with *Handlers* for the host(s)/application(s) that you will use. Select *Host List* from the *Access Point Menu* and the form shown in Figure 4-1 appears. Under TCP/IP host networks, the available handler selections are TN3270, TN5250, PTTY, TCP/IP.

```
                          Host List Setup

 Menu Name      Handler   Active                Custom Options
[VTERM       ] [VTERM   ] [yes]  [                                        ]
[PTTY        ] [PTTY    ] [yes]  [                                        ]
[SD3270      ] [3270    ] [yes]  [                                        ]
[TR3270      ] [3270    ] [yes]  [                                        ]
[TN3270      ] [TN3270  ] [yes]  [                                        ]
[SD5250      ] [5250    ] [yes]  [                                        ]
[TN5250      ] [TN5250  ] [yes]  [                                        ]
[jeff        ] [VTERM   ] [yes]  [                                        ]
[            ] [NONE    ] [no ]  [                                        ]
[            ] [NONE    ] [no ]  [                                        ]
[            ] [NONE    ] [no ]  [                                        ]
[            ] [NONE    ] [no ]  [                                        ]
[            ] [NONE    ] [no ]  [                                        ]
[            ] [NONE    ] [no ]  [                                        ]
[            ] [NONE    ] [no ]  [                                        ]
[            ] [NONE    ] [no ]  [                                        ]
[            ] [NONE    ] [no ]  [                                        ]
[            ] [NONE    ] [no ]  [                                        ]

                           <ESC>=QUIT  <ARROWS>=MOVE
```

Figure 4-1 Host List Setup Form

To enter a new selection, use the right or down <Arrow> keys to place the cursor on a blank line under the *Menu Name* column. Enter a name which will become the name of the option as shown on the RF terminal. Upper case and lower case keys can be used and are reflected in the name as it appears in the *Host List*.

Press the right <Arrow> to proceed to the *Handler* field. This is a toggle field and the <Tab> key is used to change the selection. The handler refers not only to the type of connectivity that the application or host will work under, but also to the software that processes (handle) that application/ host in the PowerNet server. A variety of handlers can be used with the TCP/IP software and choices correspond to whether terminal emulation or client streaming software was purchased.

Symbol Enabler Setup

This host kit requires a Symbol Enabler V3.0 or higher. The Enabler has two methods of operation. Please refer to the Symbol supplied documentation and ensure the Enabler is setup using the "Connect Style" interface. In later versions of the Enabler no special setup is required, and it sets up both styles automatically based on the port number used. Please ensure that the address between the TCP Host setup and the Enabler setup match.

Theory of Operation

Enabler products are host resident giving the programmer an API for RF devices. They are designed to simplify writing RF applications. Enabler products use Symbol's STEP client in the RF Terminals. This interface is considered a "client streaming" type. The data exchange between the RF terminal and the host application is by STEP dialog – a Symbol proprietary terminal interface that has provisions for controlling the RF terminal.

The application is written and controlled entirely from the host. This application controls terminal behavior by using the STEP dialog generated by the Enabler programming API. This arrangement greatly reduces the development effort for client streaming type applications since no special RF terminal programming is required. However, this method of programming does however have a steeper learning curve and development time compared to Terminal Emulation. It should therefore be reserved for those types of applications that benefit from this architecture and tolerate the longer development times.

PTTY and Other Handlers

The *PTTY Handler* and the *Other Handler* forms are almost identical. They contain fields for linking the *Host List* option with software that allows the RF terminal to establish a session with the host. The *UNIX Handler* uses the UNIX Login procedure. Other, can use protocols such as TELNET for establishing a session. The terminal software used is **tip.HEX** which is downloaded to the terminals.

Figure 4-5 shows the PTTY Handler Setup Form.

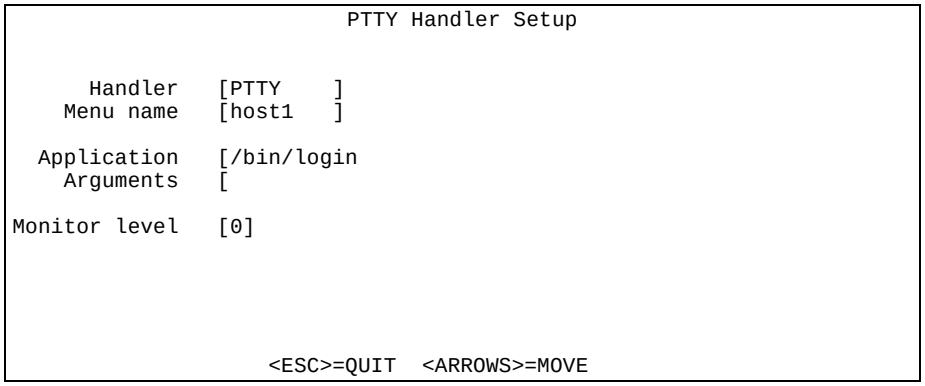


Figure 4-5 UNIX Handler Setup Form

PTTY and Other Handlers Setup Form - Fields

The fields found on these forms are described below.

Handler [Display Field] This field displays the type of *Handler* which is either *PTTY* or *Other*. It cannot be edited.

Menu Name [Display Field] This field displays the name of the host or application as it will appear in the *Host List*. This name is as entered on the *Host List Setup Form*. It cannot be changed here.

Application [Text Field] Enter the path and filename of the application which is used to establish a session either with the host or the (local session). The selections are listed in the table on Page 4-6.

Arguments [[Text Field] Enter the arguments (if any) that are passed to the application.

Monitor [Toggle Field] Use this field to control levels of information collected by the *RF Log*. The selections are 0-9. This field is not available on the *Other Handler Setup Form*.

Client Streaming Packages - TCP/IP Handler

Client streaming software uses a *TCP/IP Handler*. The setup form is shown in Figure 4-6. If more than one host will be used, a separate *TCP/IP Handler* must be created for each host.

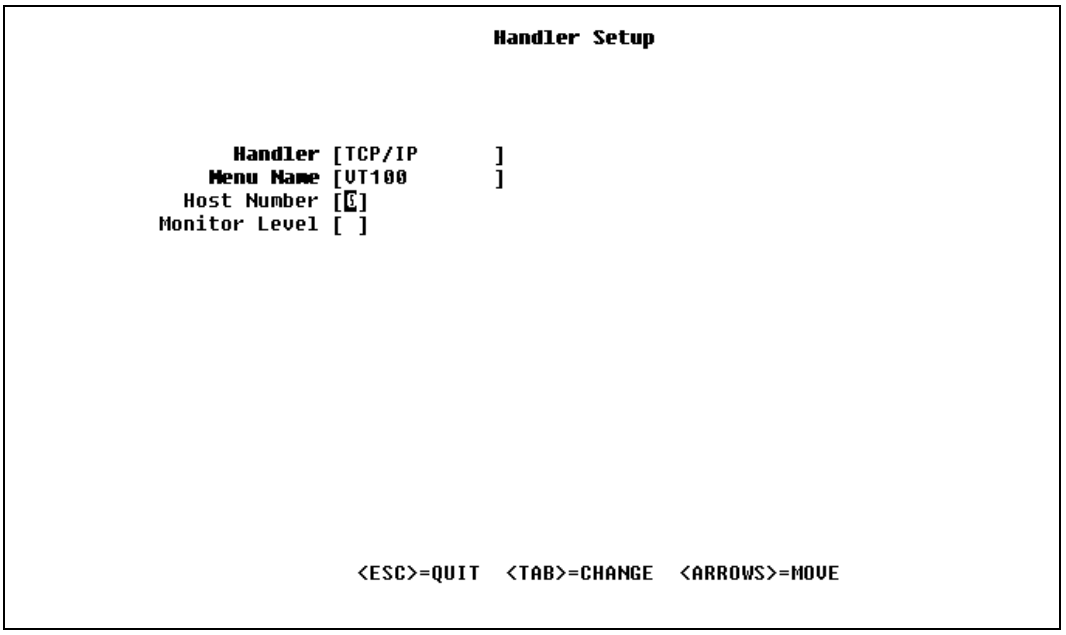


Figure 4-6 TCP/IP Handler Setup Form

TCP/IP Handler Setup Form - Fields

The fields on this form are explained below.

Handler [Display Field] This field displays the type of *Handler* which is *TCP/IP*. It cannot be edited.

Menu Name [Display Field] This field displays the name of the host or application as it appears in the *Host List*. This name is as entered on the *Host List Setup Form*. It cannot be changed here.

Host Number [Toggle Field] Select the Transaction Host (*0,1,2,3*) that is used with this handler when the name specified in the **Menu Name** field is selected from the *Host List* on the RF terminal.

Monitor Level [Toggle Field] Use this field to control levels of information collected by the **Terminal Log**. The selections are 0-9.

Completing The Handler Setup

When the *Handler Setup Form* is complete, press <Esc> to save the form and return to the *Host List Setup Form*. Use the right <Arrow> key to proceed to the *Active* field on this form. Initially, a *Host List* option is not active. When it is completely configured, use the <Tab> key to toggle the **Active** field to *Yes*. It now appears in the *Host List*, on the terminal.

Note The *Custom Options* field on this form should be left blank unless Release Notes, which can accompany the package, indicate otherwise or you wish to remap keys. See *Appendix A.* for more information.

Press <Esc> to save the *Host List Setup Form* and return to the *TCP Network Menu*.

This page intentionally left blank.

Appendix
Client Streaming API

Introduction

This appendix describes the protocols and conventions used to establish and maintain client streaming. This produces an environment of transaction-oriented sessions, between RF terminals and host applications. The client-streaming protocols utilize a pair of sessions to multiplex all terminal traffic. It is the most efficient method available for handling large numbers of terminal sessions.

This API is common across all connectivities, including TCP/IP, DECnet, X.25, Serial RS-232, etc. The API can be used with terminals running standard terminal emulation software such as CCP, STEP, VTERM, etc. It can also be used with custom client applications developed for RF terminals.

Host Assignment for RF Terminals

The PowerNet server implements the HIP protocol for host system and/or application selection by way of the *Handler*.

At a minimum, establishing a terminal session with the host requires that the terminal issue a HIP ASSIGN or ATTACH request. Refer to the SHIP protocol documentation for details.

The PowerNet server converts the host names received in HIP ASSIGN or ATTACH requests to terminal processes using the *Host List* and *Handler* options. Editing of the *Host List* is performed with the PowerNet server software and is available from the specific connectivity *Network Menu*. Changes are registered immediately without interrupting RF network operation.

Host Session

Each host interface uses two logical sessions, each in half duplex mode. One session is for transmitting data from the PowerNet server to the host (Host RX), and the other is for transmissions from the host to the PowerNet server (Host TX).

TCP/IP Host Sessions

PowerNet servers using TCP/IP connectivity are supplied with an Internet Address of 90.0.0.10 and 90.0.0.1 is used for the Host Address. The first host RX Internet ports are defaulted to 1028. Options within the specific connectivity *Network Menu* allow you to change the Internet and Port addresses and allow for a total of four hosts.

MUX Packet

The PowerNet server multiplexes each RF terminal session over the host RX session by inserting a four character header in front of the terminal data. The resulting three-field packet is [H][XXX][DATA] where H is the MUX HEADER character, XXX is the MUX ID field, and DATA is the transaction data field. APPC and TCP/IP connections provide data integrity at the data link layer and therefore, a frame check sequence is not required.

The MUX HEADER byte is used to distinguish between data and control transactions. The header byte of a data transaction record is the ASCII character, M. Polling packets, discussed later, are identified by the header byte ASCII character, P. The MUX ID field contains three ASCII digit character, 0-9, that are right-justified with leading zeros. The decimal value is generated by the PowerNet server and corresponds to a unique RF terminal address. Note that the value is mapped on the PowerNet server, so that terminal number 65 is represented as 128, number 66 as 129, and so on. The MUX DATA field may not exceed 508 bytes in length. For all host connection types, except serial, the data field is transparent to both the PowerNet server and RF network and therefore may contain binary data. Transparency over serial connections is dependent upon the protocol being used.

The MUX header must be present in the host TX transmission and will be stripped prior to transmission to the RF terminal.

Polling

The PowerNet server can be configured to generate a poll message to the host at regular intervals. This message is identified by a MUX HEADER that is the ASCII character, P. This facility can be used by host applications to implement an active link monitor.

Client Streaming Application Development in TCP/IP Environments

This section contains guidelines for developing client streaming applications for RF systems using the TCP/IP PowerNet server. In client streaming applications the host application is design specifically for RF terminals. Instead of sending

video terminal commands, these applications exchange specialized packets with the RF terminals. When correctly implemented, client streaming allows support to a greater number of terminals and higher transaction rates, at the cost of greater development effort (when compared to video terminal emulation). Client streaming can also result in longer battery life. The Figure C-1 shows a TCP/IP client streaming configuration:

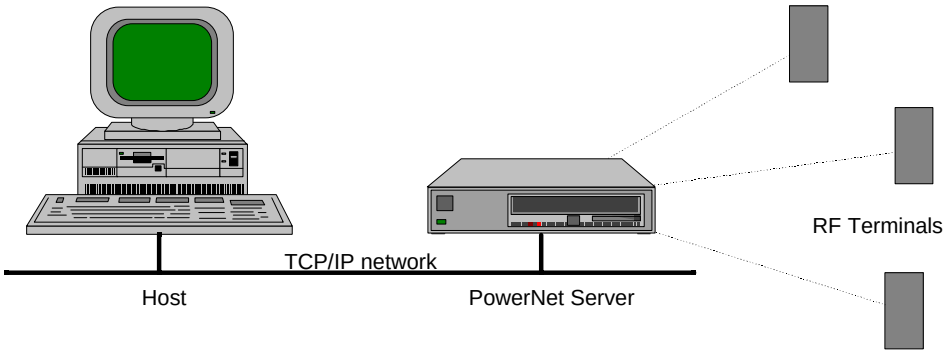


Figure C-1 TCP/IP Client Streaming Configuration

RF Terminal Software

There are two basic options for RF terminal software in a client streaming environment:

- STEP—This Symbol standard software, with commands to display messages, creates input forms, sends data to a printer connected to the terminal, etc. Since STEP is ready and tested, it can reduce the development costs of a client streaming application. On the other hand, it is generic software, with little option for doing special data processing in the terminal (i.e., input validation).

- Custom Software—With this option, the developer can optimize the RF communication for his specific application and implement specific data processing in the terminal. On the other hand, this development requires learning the RF terminal software architecture and development tools.

On most applications, STEP is recommended, unless it is unable to fulfill some basic application requirement. Should you decide to write your own RF terminal software, keep in mind that:

- A decisive factor in realizing the client streaming potential is the specification of the packets' exchange between the host application and the RF terminal. The packets' sizes should be kept minimum, do not leave wasted positions and code the data in binary.

- Battery life is significantly affected by how long the radio stays active. Lower battery consumption is achieved by working in an exclusive transactional form, alternating transmission and reception, instead of allowing unsolicited messages from the host.

The RF terminal can communicate in two modes: *datagram* and *session*. *Datagram* is more efficient, but there is no delivery guaranty; *session* gives reliable message delivery, at the cost of greater overhead. For most applications session mode is preferred.

Host Software How PowerNet servers communicate with host applications

TCP/IP allows the communication between two applications, be they in the same or in different computers. Each side in a TCP/IP communication is defined by two values:

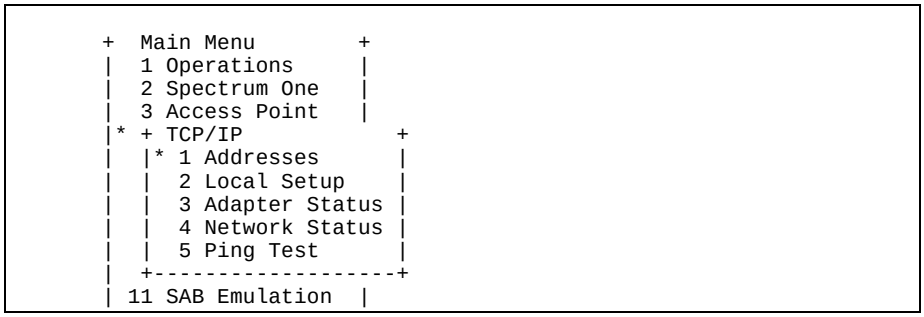
- The IP address, that defines the machine where the application is. On most computers there is a table to associate names to these addresses.

- The port number, that selects the application in a machine. Some of the port numbers are reserved for standard tasks, like remote terminal connection (Telnet), communication testing (Echo), etc.

TCP/IP defines two protocols for inter-application communication, the User Datagram Protocol (UDP) and the Transmission Control Protocol (TCP). UDP allows connectionless Datagram exchange, with no delivery guaranty. TCP gives a reliable message delivering service. he communication between the PowerNet server and the host application uses two TCP connections, one for transmission and one for reception. The messages for all the RF terminals are multiplexed over these two connections.

Host Software PowerNet Server Configuration

Before connecting the PowerNet server to the TCP/IP network, it is necessary to make a few configuration changes. All TCP/IP PowerNet server configurations are in the TCP Network menu, that is found in the Main Menu:



```
      | 12 TCP/IP-CS      |
      | 13 Serial-CS      |
      | 14 TCP/IP-STEP    |
      | 15 Hot Spare      |
      | 16 Object Editors |
      | 17 Utilities      |
      --------------------+

                  <ESC>=QUIT  <ARROWS>=MOVE  <ENTER>=SELECT
```

Figure C-2 Main Menu

The first step is to configure the TCP/IP Network Addressing option with the name and address of the involved machines:

```
                        TCP-IP Network Addresses

       Name             Address                 Name             Address
[ncu              ][90.0.0.1       ]     [                ][                ]
[winnt            ][90.0.0.2       ]     [                ][                ]
[                 ][               ]     [                ][                ]
[                 ][               ]     [                ][                ]
[                 ][               ]     [                ][                ]
[                 ][               ]     [                ][                ]


                              <Esc> = Quit         <Arrows> = Move
```

Figure C-3 TCP-IP Node Addresses

In the above screen we defined two machines, the PowerNet server, named *ncu* with IP address 90.0.0.1 and the host, named *winnt* with IP address 90.0.0.2. Next, configure the NCU parameters in the PowerNet server addressing option:

```
                            Local TCP/IP Setup

                     Interface 0           Interface 1
              State [ON ]                 [OFF]
            Adapter [sme0 ]               [     ]
          Node Name [trn2001p        ]    [                 ]
            Address [206.183.67.183 ]     [                ]
            Netmask [255.255.255.128]     [                ]
          Broadcast [206.183.67.255 ]     [                ]
        Router Name [                ]    [                 ]
         Token Ring
      Source Routing [NO ]                [NO ]
          Ring Speed [16]                 [16]
               Media [UTP]                [UTP]

                  <ESC>=QUIT  <SPACE>=CHANGE  <ARROWS>=MOVE
```

Figure C-4 PowerNet server TCP-IP Setup

The next step is to define the TCP connections that will be used. The PowerNet server allows up to four TCP connections, defined in the TCP/IP-CS Setup option. This option is accessed through the TCP/IP Setup screen. Return to the Main Menu and select TCP/IOP-CS.

```
                      TCP/IP-CS Setup

                                  Host     PowerNet  Log
       State  Host Node Name     IP Port  IP Port  Level  Poll
Host 0 [ON ] [winnt           ]  [1029 ]  [1028 ]  [0]   [0    ]
```

```
Host 1 [OFF] [ttysrv          ]  [1028 ]  [1029 ]  [0]   [0    ]
Host 2 [OFF] [ttysrv          ]  [1028 ]  [1030 ]  [0]   [0    ]
Host 3 [OFF] [ttysrv          ]  [1028 ]  [1031 ]  [0]   [0    ]

             Host 0       Host 1       Host 2       Host 3
 Record Format [DELIMITED]  [DELIMITED]  [DELIMITED]  [DELIMITED]
     Delimiter [124]        [124]        [124]        [124]
```

Figure C-5 Transaction Port Setup

In the above screen there is only one TCP connection active (state = on), with the following characteristics:

- The host application is in the *winnt* machine and waits for calls from the PowerNet server on port 1029.
- The PowerNet server waits for call from the host application on port 1028.

Finally, it is necessary to create a host list entry for the application:

```
                          Host List Setup

 Menu Name      Handler   Active                Custom Options
[WHIP-CHECK  ] [CHECK   ] [no ]  [                                    ]
[STEP-CHECK  ] [CHECK   ] [yes]  [                                    ]
[tcp-cs      ] [TCP/IP  ] [yes]  [-cm -slp -test                      ]
[            ] [NONE    ] [no ]  [                                    ]
[            ] [NONE    ] [no ]  [                                    ]
```

Figure C-6 Host List Setup

In the above screen, the entry `tcp-cs` is the one that used to connect to the host application. Access this screen by returning to the Main Menu and selecting the Spectrum One option and then the Host List option. The options in the Custom Options field have the following meanings:

- **-cm**—Obligatory with TCP/IP Client Streaming, assures that some commands from the RF driver do not go to the TCP/IP driver.
- **-slp**—Stops SLP headers from being sent to the host application. The SLP protocol is used by the Symbol RF terminals only when in Session mode.
- **-test**—Makes the PowerNet server respond to the “host ready” test made by the SLP protocol, speeding up the RF terminal connection to the host application.

In the Handler Setup for the TCP/IP entry we select which host, defined in the Transaction Ports screen, is used:

```
                          Local TCP/IP Setup

                   Interface 0           Interface 1
             State [ON ]                 [OFF]
           Adapter [sme0 ]               [     ]
         Node Name [trn2001p        ]    [                ]
           Address [206.183.67.183 ]     [                ]
           Netmask [255.255.255.128]     [                ]
         Broadcast [206.183.67.255 ]     [                ]
       Router Name [                ]    [                ]
        Token Ring
    Source Routing [NO ]                 [NO ]
        Ring Speed [16]                  [16]
             Media [UTP]                 [UTP]
```

Figure C-7 Handler Setup

The diagram below shows the relationship between all these configurations:

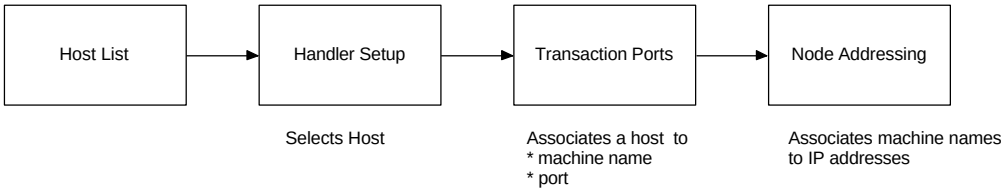


Figure C-8 TCP/IP Configuration

Host Software Communication between PowerNet server and host applications

The communication between the PowerNet server and the host application uses two TCP connections, one for transmission and one for reception. The establishment of these two TCP connections is as follows:

- The PowerNet server gets from the Transaction Port Setup the host name, and converts it to an IP address, using the Node Addressing. The host application port number is also gotten from the Transaction Port Setup.

- The PowerNet server tries to establish a connection to the host application, with this IP address (that selects the host machine) and port number (that selects the application in the host).

- The host application, when it detects the first connection, will then establish the second connection, using the PowerNet server's IP address and port number. The first connection will be used to sent messages from the PowerNet server to the host application and the second connection will be used the other way.

The diagram below shows the connections for the configuration screens shown before. Note that, just like in phone connections, callers need to know the "number" (IP address and port number, in our case) of whom they are calling, but

not the other way around. So, the port numbers indicated by xxx and yyy are irrelevant.

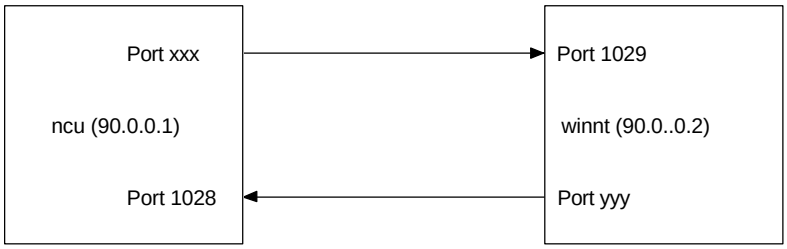


Figure C-9 TCP/IP Port Connection

Host Software Format Exchanged by PowerNet server and Host Application

As before, the messages exchanged by the host application and the many RF terminals are multiplexed over the two TCP connections. This is done by a three digit number that identifies the RF terminal that sent or will receive the message. The PowerNet server generates this number by subtracting 49 from the RF address of the terminal.

At the end of each message a delimiter must be appended. The default value for this delimiter is '|' (ASCII code 124) and it cannot be used inside the message. This delimiter can be changed to any other ASCII character by the inclusion of the option **-er xxx** (where xxx is the decimal ASCII code of the character) in the *Custom Options* field in the *Host List*.

The application in the RF terminal receives and sends messages without a header and delimiter. For example, to send the message ABC to a RF terminal with address 65, the following message should be sent to the PowerNet server:

```
M 0 1 6 A B C |
```

It is important to note that the RF terminal address, and how it is converted in the three digit number, is irrelevant to the application. When a terminal connects to the RF network, it should send a *Logon Ready* message to the application. When it receives this message, the application will store the number generated by the PowerNet server and use it to send messages to the terminal and to recognize the messages it sent.

Host Software Using Sockets Interfaces in Host Application

- Create (using the socket function) two sockets, one to wait for the PowerNet server call and the other to establish the transmission connection.

- Associate (using the bind function) the first socket to the local address (IP address + port number).

- Associate a queue (using the listen function) to this socket and wait for a connection (using the accept function) from the PowerNet server.

When a connection from the PowerNet server is received, the accept function returns a new socket that will be used to receive messages from the PowerNet server. Associate (using the bind function) the second socket to the local address (IP address + port number) and connect (using the connect function) to the PowerNet server. From this point, the application can start to receive (using the read function) and send (using the write function) messages.

Before exiting, the application should close (using the close function) all the sockets.

Host Software Example

This example was developed under Windows/NT, using the WinSock interface. It assumes that the terminal is running STEP.

```
#include <stdio.h>
#include <conio.h>
#include <ctype.h>
#include <stdlib.h>
#include <winsock.h>

void main (int argc, char *argv[]);

// STEP command buffers
// & will be replaced by ESC
char buf_gret[] = "&C00&D0401020Greating from TCP/IP&I0510001|";
char buf_loff[] = "&L|";

void main (int argc, char *argv[])
{
  WSADATA wsaData;
  WORD wVersionReq = 0x0101;
  SOCKET s_listen, s_snd, s_rcv;
  SOCKADDR_IN addr, ncuaddr;
  IN_ADDR ncuIn;
  int sizncuaddr;
  int i;
  int nrec;
  char buf_rx [512];
  char buf_tx [512];

// Prepares STEP command buffers
  for (i = 0; i < sizeof(buf_saud); i++)
    if (buf_gret[i] == '&')
     buf_gret [i] = 0x1B;
  for (i = 0; i < sizeof(buf_loff); i++)
    if (buf_loff[i] == '&')
     buf_loff [i] = 0x1B;

  // Tests if arguments where given
  if (argc != 3)
  {
    printf ("Usage: tcpcs ncu_port host_port\n");
```

```
    exit (1);
  }

  // Starts WinSock
  if (WSAStartup (wVersionReq, &wsaData) != 0)
  {
    printf ("Error %d in startup!\n", WSAGetLastError ());
   exit (2);
  }
  printf ("WinSock %s initialized.\n", wsaData.szDescription);

// Creates two sockets
  s_listen = socket (AF_INET, SOCK_STREAM, 0);
  s_snd = socket (AF_INET, SOCK_STREAM, 0);
  if ((s_listen == INVALID_SOCKET) || (s_snd == INVALID_SOCKET))
  {
    printf ("Error in socket creation!\n");
   if (s_listen != INVALID_SOCKET)
     closesocket (s_listen);
   if (s_snd != INVALID_SOCKET)
     closesocket (s_snd);
    WSACleanup ();
    exit (3);
  }
  printf ("Sockets were succeffuly created.\n");

  // waits for NCU call
  addr.sin_family = AF_INET;
  addr.sin_port = htons ((u_short) atoi (argv[2]));
  addr.sin_addr.s_addr = htonl (INADDR_ANY);
  if (bind (s_listen, (LPSOCKADDR) &addr, sizeof(addr)) ==
SOCKET_ERROR)
  {
    printf ("Error %d in bind!\n", WSAGetLastError());
    closesocket (s_listen);
    closesocket (s_snd);
    WSACleanup ();
    exit (4);
  }
  printf ("Bind OK.\n");
  if (listen (s_listen, 1) == SOCKET_ERROR)
  {
    printf ("Error %d in listen!\n", WSAGetLastError());
    closesocket (s_listen);
    closesocket (s_snd);
    WSACleanup ();
    exit (5);
  }
  siznuaddr = sizeof (ncuaddr);
  s_rcv = accept (s_listen, (LPSOCKADDR) &ncuaddr, &siznuaddr);
  if (s_rcv == INVALID_SOCKET)
  {
    printf ("Error %d in accept!\n", WSAGetLastError());
    closesocket (s_listen);
    closesocket (s_snd);
    WSACleanup ();
    exit (6);
  }
  memcpy (&ncuIn, &ncuaddr.sin_addr.s_addr,4);
  printf ("Received call from NCU (IP=%s, port=%d).\n",
          inet_ntoa(ncuIn), ntohs(ncuaddr.sin_port));
// makes second conection with the NCU
  addr.sin_family = AF_INET;
  addr.sin_port = 0;
  addr.sin_addr.s_addr = htonl (INADDR_ANY);
  if (bind (s_snd, (LPSOCKADDR) &addr, sizeof(addr)) == SOCKET_ERROR)
  {
    printf ("Error %d in second bind!\n", WSAGetLastError());
```

```
      closesocket (s_listen);
      closesocket (s_snd);
      closesocket (s_rcv);
      WSACleanup ();
      exit (7);
    }
    printf ("Second bind OK.\n");
    addr.sin_family = AF_INET;
    addr.sin_port = htons ((u_short) atoi (argv[1]));
    addr.sin_addr.s_addr = inet_addr (inet_ntoa (ncuIn));
    if (connect (s_snd, (LPSOCKADDR) &addr, sizeof(addr)) ==
  SOCKET_ERROR)
    {
      printf ("Error %d in connect!\n", WSAGetLastError());
      closesocket (s_listen);
      closesocket (s_snd);
      closesocket (s_rcv);
      WSACleanup ();
      exit (8);
    }
    printf ("Connect OK.\n");
  // Loop to talk with the RF terminals
    while (!_kbhit())
    {
      nrec = recv (s_rcv, buf_rx, sizeof(buf_rx), 0);
      if (nrec == SOCKET_ERROR)
      {
        printf ("Error %d in reception!\n", WSAGetLastError());
        closesocket (s_listen);
        closesocket (s_snd);
     closesocket (s_rcv);
        WSACleanup ();
        exit (8);
      }
      else if (nrec == 0)
      {
        printf ("Conection closed!\n");
        closesocket (s_listen);
        closesocket (s_snd);
     closesocket (s_rcv);
        WSACleanup ();
        exit (8);
      }
      else
      {
       printf ("Message received:\n");
       // list buffer
       for (i = 0; i < nrec; i++)
       {
         if (isprint (buf_rx [i]))
              printf ("%c  ", buf_rx [i]);
           else
             printf ("%2.2X ", buf_rx [i]);
           if ((i & 15) == 15)
             printf ("\n");
       }
       if (i != 0)
         printf ("\n");

       printf ("Sending answer: ");
       memcpy (buf_tx, buf_rx, 4);         // Mxxx
       if (buf_rx [5] == 'L')              // Logon Ready
       {
         printf ("Greatings.\n");
           memcpy (buf_tx+4, buf_saud, sizeof (buf_saud)-1);
           send (s_snd, buf_tx, 3+sizeof(buf_saud), 0);
       }
       else
```

```
        {
          printf ("Log off.\n");
              memcpy (buf_tx+4, buf_loff, sizeof (buf_loff)-1);
              send (s_snd, buf_tx, 3+sizeof(buf_loff), 0);
        }
       }
     }

     // final clean-up
     getch ();
     closesocket (s_listen);
     closesocket (s_snd);
     closesocket (s_rcv);
     WSACleanup ();
     printf ("Normal End.\n");
     exit (0);
   }
```

Note: In this example WinSock blocking functions were used. In a practical application it is recommended the use of the Asynchronous (non-blocking) functions, that will not block the application until the communication operation is concluded.

Conclusion

With client streaming, it is possible to create high performance RF applications that support a large terminal population with high transaction rates. Even though client streaming application development is more complex than simple video terminal emulation, it requires only a limited knowledge of RF terminals and TCP/IP as well as following of some basic guidelines.

Reference Materials

Spectrum One Development System - Application Programmer's Guide, Symbol Technologies, 1992

Series 3000 STEP - Application Programmer's Guide, Symbol Technologies, 1992

Network Controller Interface - Programmer's Guide, Symbol Technologies, 1991

Programming WinSock, Arthur Dumas, Sams Publishing, 1995