PAC-COMM

TNC 220

REFERENCE MANUAL

[redacted]

REFERENCE MANUAL

for the

Pac-Comm HF/VHF Terminal Node Controller

Model TNC-220

Pac-Comm Packet Radio Systems, Inc.
3652 W. Cypress Street
Tampa, FL 33607-4916

Technical Support and Business Office (813) 874-2980
Electronic Mail System (813) 874-3078
Telefax (813) 872-8696
TELEX 650-288-1526 MCI

The information contained in this document has been carefully checked and is believed to be entirely reliable, however, no responsibility is assumed for inaccuracies. Pac-Comm Packet Radio Systems, Inc. (Pac-Comm) reserves the right to make changes to any products or documentation for any purpose including improvement of reliability and functionality without obligation to purchasers of previous equipment. Pac-Comm does not assume any liability arising out of the application or use of any product or circuit described herein; neither does it convey any license under its patent rights or the rights of others.

The equipment described in this manual must be installed and operated in accordance with the printed procedures and should NOT be used for any purpose which involves the safety of life or limb.

Second Edition, First Printing - June 1987

REFERENCE MANUAL CONTENTS

F I G U R E S

T A B L E S

A P P E N D I C E S

Chapter 1

INTRODUCTION

Welcome to the exciting world of amateur packet radio! You have joined the ranks of the fastest growing mode in amateur radio. The purpose of the Pac-Comm Packet Radio Systems, Inc. (Pac-Comm) Terminal Node Controller model TNC-220 is to act as an interface between your ordinary voice radios, such as a 2-meter FM transceiver and an HF SSB transceiver, and your computer. The TNC-220 will perform all of the "magic" of establishing error-free communications between your station and another packet-radio equipped station. You will be able to have a "private channel" while sharing a frequency with other packet stations, "read the mail" of other QSOs, operate remote computer "bulletin board" or "mailbox" stations, handle message traffic -- in short, be able to enjoy all the advantages of digital communication techniques in your ham shack.

Your Pac-Comm TNC-220 is the key to your packet station. It is a new TNC design based on the TAPR TNC 2 which was in turn based on the original TAPR TNC and inherits all of the advanced features of those designs. The design of the TNC-220 includes the latest features and improvements requested by the thousands of TNC-200 owners and suggestions from hams all over the world.

Figure 1-1. TNC-220 (with tuning indicator option) front and rear views.

If you have purchased a bare board or kit, the ASSEMBLY APPENDIX will take you step by step through the construction and testing of your TNC-220. The REFERENCE MANUAL will explain how to connect your TNC-220 to your station computer and interface the TNC-220 to your radios. The REFERENCE MANUAL contains a detailed breakdown of the various commands the TNC will accept and messages it may report. The manual also contains a description of the hardware design of TNC-220 and troubleshooting hints. The manual concludes with an overview of packet radio protocol. The **COMMAND REFERENCE CARD** contains a complete listing of all TNC-220 commands in a convenient form for reference while you are operating.

NOTE: Please be sure to read any Errata, manual changes or software notes that accompany your manual and TNC-220. These documents will describe all changes and upgrades made to your unit. Pac-Comm policy is to make improvements to the TNC-220 hardware and software as soon as production factors will permit.

While there are no lethal voltages present in the TNC-220, the user should exercise due caution when connecting the unit to power sources and radio equipment to insure that no hazardous situation is created through the interconnection. Carefully inspect for any potential differences in the ground circuits of equipment powered from different sources, then fully ground all units to a solid earth ground.

When operating the TNC-220 out of its cabinet, use caution to avoid touching the voltage regulator and the associated 5 ohm resistor located at the corner of the TNC-220 circuit board near the power connector, as these units become hot during operation. Also be sure to comply with instructions about the regulator heatsink before operating the unit out of its cabinet to avoid damage to the unit.

Every effort has been made to make the supplied documentation as accurate as possible; however, Pac-Comm will not assume responsibility for any damages incurred. Pac-Comm reserves the right to change documentation or equipment at any time without notice.

PAC-COMM WARRANTIES

ASSEMBLED AND TESTED UNITS

Pac-Comm Packet Radio Systems, Inc. (Pac-Comm) warrants "Assembled and Tested" amateur line products to be free from defects in materials and workmanship for a period of one year from the date of purchase by the retail customer. During the warranty period, Pac-Comm will, at its option, repair, replace, or refund the purchase price of any unit that it determines to be defective. This warranty does not apply to any product which has been modified, damaged through carelessness, misuse or neglect, or which has been repaired outside Pac-Comm facilities without permission. Consequential damages are not covered.

ELECTRONIC KITS

Pac-Comm Packet Radio Systems, Inc. (Pac-Comm) warrants "Electronic Kits" to be composed entirely of components and materials which are free from defects in materials and workmanship and appropriate for the function to be performed. This warranty shall extend for a period of thirty days from the date of purchase by the retail customer. Pac-Comm will, at its option, replace all defective or missing components, or provide installation of the components without charge. Pac-Comm reserves the right to require the return of any defective component for inspection. This waranty does not apply to any component which has been modified or damaged through carelessness, misuse, or neglect. Consequential damages are not covered.

SOFTWARE

Pac-Comm Packet Radio Systems, Inc. (Pac-Comm) warrants "Software" to meet the advertised specifications and be suitable for the advertised purposes. The medium on which the software is delivered to the retail customer is warrantied at the time of delivery to be free of defects which would impair the performance of the software. Pac-Comm will replace the software medium, if the medium is returned to Pac-Comm within 30 days and Pac-Comm's examination finds the medium to be defective.

REPAIR SERVICE

Pac-Comm provides a quick repair service for our products, whether in or out of warranty. Shop rates are modest. Carefully pack the unit and send it to the Pac-Comm address with a written description of the problems experienced and the repair service desired. Normal service turnaround is under five days. We stock all parts for our units. Call for your needs.

Chapter 2

COMPUTER INTERFACING

The TNC-220 communicates with your computer through a serial port using either RS-232C or ´TTL´ level signals. One or the other of those signal levels should properly interface to most any computer available today. Since there are so many computers on the market today, it is impractical for this chapter to provide detailed instructions for each computer. See the REFERENCE MANUAL Appendices for specific computer interfacing information. Throughout this manual we will use the term "computer" to refer to the computer or terminal you use to communicate with your TNC-220.

Serial Port Signals

The serial port connector on your TNC-220 is on the rear panel and is marked "Terminal." The pins on the serial port connector of TNC-220 that must be connected are shown in Table 2-1. Note that in the RS-232C mode the TNC-220 connects to a computer exactly as if the TNC-220 were a standard RS-232C modem. If you have successfully used your computer with a telephone modem, hook it up to TNC-220 in the same way. Use whatever program you ordinarily use to communicate with the modem and proceed to the section, **Verifying Serial Port Operation**. If you plan to use the TTL level connection to your computer see the section, **Serial Port Operation at TTL Levels.**

Table 2-1. RS-232C Serial Port Signals Required by the TNC-220.

Pin	Signal Name	Description
1	Frame Ground	Protective ground circuit
2	Transmit Data	Serial data from your computer to the TNC-220.
3	Receive Data	Serial data from the TNC-220 to your computer.
7	Signal Ground	The common ground for both data lines.

Some computers require an adapter card to incorporate the circuitry necessary to add a serial port to the computer. Some popular models in this category are the Apple II series, the IBM Personal Computer, many Radio Shack computers. If your computer has a 25-pin RS-232C serial port, refer to the section Computers with 25-pin RS-232C Ports. Otherwise refer to the section Computers with Nonstandard Serial Ports. Some computers have no serial port and no adapter is commercially available. Such computers are not suitable for use with TNC-220.

Computers with 25-pin RS-232C Ports

You should consult your computer manual or accessory manual to see which pins your computer uses to send and receive data on, as well as which pin is used for signal common. Follow the computer manufacturer´s recommendations for connecting the serial port to a modem. You may also find the technical information in this section useful.

Your TNC-220 is configured as Data Communications Equipment (DCE), the technical term for an RS-232C modem. Most computers are configured as Data Terminal Equipment (DTE). If this is the case for your computer, you will probably be able to simply wire pin 1 of the TNC-220 connector to pin 1 of your computer´s RS-232C port, pin 2 to pin 2, pin 3 to pin 3 and pin 7 to pin 7. You can provide these connections with a standard 3-wire (plus frame ground) male-to-female or male-to-male RS-232C extension cable, depending on whether your computer has a DB25S or DB25P connector.

If your computer is configured as DCE, you will have to wire pin 2 of your TNC-220 to pin 3 of the computer connector, and pin 2 of the computer connector to pin 3 of your TNC-220. Pin 7 of the computer connector will still connect to pin 7 of your TNC-220 serial port.

Some computers may require that pin 5 of the computer serial port connector be connected to an appropriate signal. Others may require connections for pin 8 and pin 20. You can use the computer´s output signals on pins 4 and 6 as shown in Fig. 2-3.

Computers with Nonstandard RS-232C Serial Ports

Computers with nonstandard RS-232 serial ports must meet the following conditions.

1. The TNC-220 requires that RS-232C voltage levels sent from the computer be greater than about +3 volts in one state and less than about -3 volts in the other state.
2. The polarity of the RS-232 signals must conform to the RS-232C standard. This means that the low voltage state must correspond to a logical "1" and the high voltage state to a logical "0".

```
TNC-220          Computer
   1 ------------ 1
   2 ------------ 2
   3 ------------ 3
   7 ------------ 7
               |---- 4
               |---- 5
        |  --------- 6
        |----------- 8
        |  --------- 20
```

Fig. 2-1. Serial port wiring with jumpers for auxiliary signals.

Make or buy a cable that provides the following connections. The computer serial port common pin must be tied to the TNC-220 serial port connector pin 7. The data line that sends data from the computer must be tied to the TNC-220 connector pin 2. The pin on which your computer receives data on must be tied to the TNC-220 connector pin 3. If your computer requires any other signals, you must arrange to provide them. The documentation provided with your computer or its accessory serial port should clarify any special requirements.

Serial Port Operation At TTL Levels

The TNC-220 is designed to operate with computers which have a TTL level serial port without requiring the use of an RS-232C adaptor. This includes such popular computers as the Commodore Vic-20, C-64, and C-128. A jumper on the TNC-220 circuit board may be positioned to support either RS-232C or TTL operation. TTL signals are routed through the 25 pin RS-232 connector using pins that are designated as ´Secondary RS-232C´ connections. TTL signal levels must be greater than +2.4 volts in one state and less than about +.8 volt in the other state. The following pin definitions apply:

Table 2-2. TTL serial port signals required by TNC-220.

Pin	Signal Name	Description
1	Frame Ground	Protective ground circuit
7	Signal Ground	The common ground for both data lines.
14	Transmit Data	Serial data from your computer to the TNC-220.
16	Receive Data	Serial data from the TNC-220 to your computer.

If the computer using TTL interface signals requires hardware handshaking, reference Table 2-3 and Appendix A to design the cable. See also the section entitled TNC-220 Serial Port Pin Functions.

Software Requirements

Any software package that enables your computer to act as an ASCII terminal with an ordinary telephone modem should work with your TNC-220. If you have a program that you have used successfully with a telephone modem and that you are familiar with, use that program to communicate with your TNC-220.

Verifying Serial Port Operation

Turn off the power to your computer and to your TNC-220. Connect the computer to the TNC-220 with a properly configured serial cable. Turn on your computer and start the terminal program. Follow the directions for the program you are using to set the computer's baud rate to 1200 bauds, 7 bits even parity and 1 stop bit.

NOTE: The serial port data rate used by the TNC-220 to communicate with the computer is fixed at 1200 bauds until the BAUD command is used to alter the data rate.

Turn on your TNC-220. You should see the sign-on message printed on your computer screen. This demonstrates the ability of your computer to accept data from your TNC-220. If you see nothing, switch off your TNC-220 for a few seconds, then on again. If you still see nothing, verify your wiring and restart your terminal program. If you see gibberish on your screen you should verify that you have set the computer to the proper 1200 baud rate for the TNC-220. When you have successfully read the sign-on message from your TNC-220, type **DISPLAY** followed by a carriage return. You should see a lengthy list of items on your screen. This verifies the ability of your TNC-220 to accept and respond to input from your computer. Once the TNC-220 is communicating with the computer, the terminal baud rate may be changed from the keyboard bu using the BAUD command. The terminal baud rate will remain in effect as long as the battery-backed RAM parameters remain intact.

If your TNC-220 fails to properly communicate with your computer, see the paragraph, Serial Port Configuration in Chapter 4 and the troubleshooting section in Chapter 8.

TNC-220 Serial Port Pin Function

This section describes the pins used on the TNC-220's serial port connector. It is intended for packet operators with special applications requiring hardware handshaking. This information should not be needed by most users.

Table 2-3. Serial Port Pin Designations

RS-232C			TTL		
Pin	Mnemonic	Name	Pin	Mnemonic	Name
1	FG	Frame Ground	1	FG	Frame Ground
2	TXD	Transmit Data	14	STXD	Secondary Transmit Data
3	RXD	Receive Data	16	SRXD	Secondary Receive Data
4	RTS	Request To Send	19	SRTS	Secondary Request To Send
5	CTS	Clear To Send	13	SCTS	Secondary Clear To Send
6	DSR	Data Set Ready	--	SDSR	Secondary Data Set Ready
7	SG	Signal Ground	7	SG	Signal Ground
8	DCD	Data Carrier Detect	12	SDCD	Secondary Data Carrier Detect
20	DTR	Data Terminal Ready	--	SDTR	Secondary Data Terminal Ready

Since the RS-232 pins are defined with respect to Data Terminal Equipment (DTE) (meaning the computer) end of the circuit, and the TNC-220 is wired as Data Communications Equipment (DCE) (meaning a modem) most of the standard pin names used will appear to be the reverse of the TNC-220 function. The TNC-220 serial interface is virtually identical with the TNC-200 interface.

Frame Ground is provided for attachment to the chassis of the TNC-220 and the chassis of the attached computer or terminal.

Transmit Data is an input line to the TNC-220 on which the attached device sends data.

Receive Data is an output line from the TNC-220 on which the attached device receives data.

Request to Send is an input to the TNC-220 signaling that the attached device is ready to accept data from the TNC-220. This line is connected to DTR (20) inside the TNC-200 and either may be used for hardware flow control.

Clear To Send is an output from the TNC-220 signaling the attached device to send or refrain from sending data to the TNC-220. This line is used for hardware flow control.

Data Set Ready is an output from the TNC-220 telling the attached device that the TNC-220 is operational.

Signal Ground is the common, or return, path for all signals between the TNC-220 and the attached device.

Data Carrier Detect is an output from the TNC-220. As normally configured, DCD reflects the status of the CON LED: It is true when an AX.25 connection exists between your TNC-220 and another station; it is false when no connection exists. This configuration is useful when the TNC-220 is used with a telephone style Bulletin Board system, since the AX.25 connection, analogous to a modem signal on the telephone, indicates the presence of a user. This line may be disabled or jumpered high if so required by the terminal.

Data Terminal Ready is an input to the TNC-220 signaling that the attached device is ready to accept data from the TNC-220. This line is connected to RTS (4) inside the TNC-200 and either may be used for hardware flow control.

Chapter 3
RADIO INTERFACING

Interfacing the TNC-220 to your radios involves connecting the following signals at J1 and J2. The pinouts of J1 and J2 are shown in Fig. 3-1. Both J1 and J2 use the same style of connector. Only one is illustrated.

Pin 1 Microphone audio, from the TNC-220 to your transmitter.

Pin 2 Ground, audio and PTT common.

Pin 3 Push-to-talk, to allow the TNC-220 to key your transmitter.

Pin 4 Receive audio, from your receiver to the TNC-220.

Pin 5 (J1 ONLY) FSK keying line to allow the TNC-220 to directly key an FSK HF transceiver.

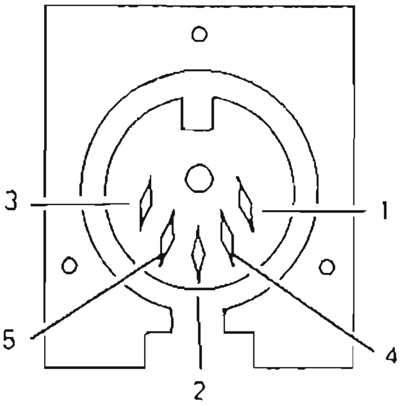


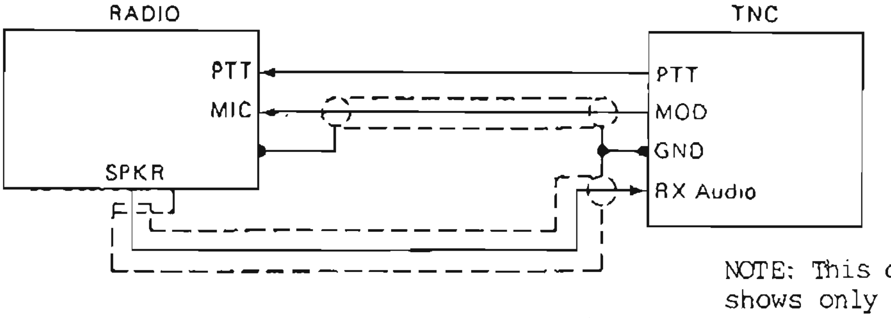
Fig. 3-1. J1/J2 radio port connector

This chapter describes how to connect these signals between your TNC-220 and your radio and how to adjust the receive and transmit audio levels appropriately. The interconnection should be planned so as to minimize pickup of stray audio and RF noise by the lines.

This chapter assumes that you are using an FM VHF radio and a SSB HF radio. If you are operating both ports on VHF (or both on HF), the information is still applicable. Two interfacing methods are presented. You will have to use the second method if you can't adjust the audio levels properly with the first method. They require no special test equipment for adjusting the audio levels. If test equipment is available, however, it should be used as described. You will need a second receiver in your shack that you can use to listen to your own signal. Read the remainder of this chapter carefully before starting to interface your TNC-220 to your radio.

Method 1: Direct Connection to Microphone and Speaker

The TNC-220 was designed to allow hookup and testing to be done without any modifications to the radio or any signal level balancing devices in the cables.



NOTE: This diagram shows only one port.

Fig. 3-2. Method One Interconnection.

For Method 1, shown in Fig. 3-2, the TNC-220's two audio signals will be fed directly into the microphone connector or similarly connected auxiliary jack of each radio, and the output of each port will be adjusted to give a proper modulation level. The receiver audio will be taken from an auxiliary audio output or speaker jack on each radio and fed directly to the TNC-220. This method is expedient but, unless you have an auxiliary audio output (most modern transceivers do), it doesn't allow you to monitor the channel or to conveniently use the rig on voice. Many packeteers devote one or more radios exclusively to packet and this direct hookup will work well in that situation. If you share your radio between packet and voice, you may want to use the second interface method for your permanent station interface, after the initial testing phase.

Remove the TNC-220 from its cabinet by removing the two screws securing the **REAR** end plate and bezel. Then remove the screw securing the regulator to the cabinet. Slide the TNC-220 out of its cabinet.

WARNING: IF THE TNC-220 IS OPERATED OUTSIDE THE CABINET FOR EVEN A FEW SECONDS, THE 7805 VOLTAGE REGULATOR MUST BE PROPERLY HEATSINKED TO PREVENT FAILURE!! The 7805 regulator must dissipate up to 3.5 watts of electrical power as heat. This is well within the normal ratings for the device IF properly heatsinked. Normally the 7805 is firmly connected to the TNC-220 cabinet for heat transfer. Use the paper clasp provided for a temporary heatsink. It is suggested that a large screwdriver blade be slipped through the 'handles' of the paper clasp to augment the heat dissipation for periods beyond approximately 5 minutes.

Connect your TNC-220 and radios as shown in Fig. 3-2. Turn on your TNC-220 and computer and start your terminal program. Connect the radios to dummy loads and listen to the transmission with another nearby radio.

Since the TNC-220 interfaces to two radio transceivers, you will need to go through the following procedure separately for each radio. Use the "PORT 2" command to cause the TNC-220 to select the VHF radio port and follow the instructions below. Then use "PORT 1" to select the HF port and repeat the procedure. Refer to the TNC-220 board layout diagram for the location of the trimpots to be adjusted.

1. Type in the command CAL <CR> and type the letter "K" to cause the TNC-220 to key the selected radio with a steady tone.

2. With the TNC-220 keying the transmitter, adjust the transmit audio level trimpot for the port you are using while you listen to the monitoring receiver. Turn the adjustment screw on the trimpot until no increase in output level is heard at the monitoring receiver.

3. Then rotate the adjustment screw of the trimpot in the opposite direction until the audio signal on the monitoring receiver is slightly, but noticeably, reduced from the maximum level.

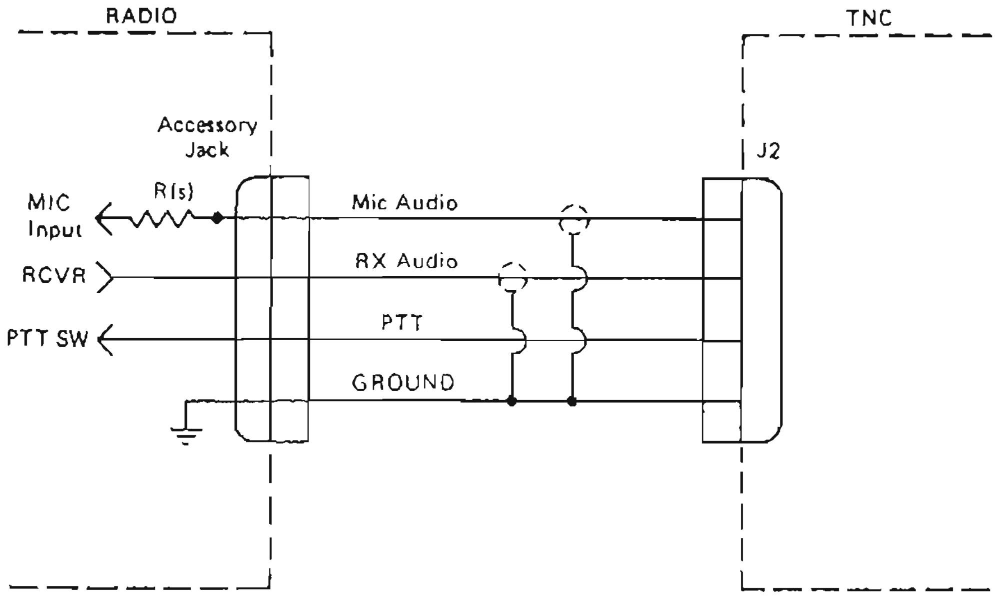
4. The transmitter deviation is now set to approximately the correct level. Type the command "K" to unkey the transmitter.

5. The TNC-220 is designed to operate optimally with a 200 to 700 millivolt audio input signal. This is the signal level provided by most modern transceivers at the "Fixed Audio Output" connector on the rear panel. It is recommended that a fixed audio output be used if available. If you use the headphone or external speaker jack on your radio, adjust the audio level to approximately 500 mV with the aid of a high impedance voltmeter or oscilloscope. If no such measuring equipment is available, set the volume control at the position normally used for comfortable listening. This should provide an adequate signal output. NOTE: The DCD LED on the TNC-220 is software controlled and is not suitable for setting audio input levels as is done on the TNC-200.

If you notice a significant hum level in the monitored audio in Step 3, take measures to remove it. This may require shielded wire (recommended in any event) in your microphone audio circuit. If your transmitter has an adjustable microphone gain control, try reducing the sensitivity of the transmitter microphone circuit and increasing the signal level from your TNC-220 to minimize hum or other noise problems.

Method 2: Accessory Jack or Interface Box Connection

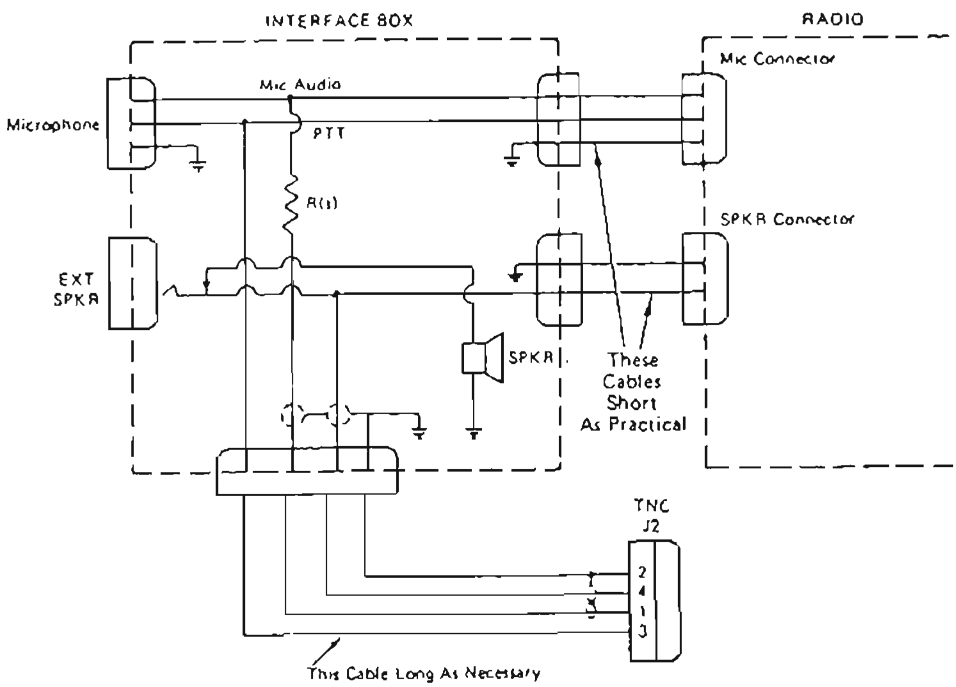
If your radio has an accessory jack with PTT, transmit audio, and receive audio signals, all connections between the radio and TNC-220 may be done through this jack. If the audio output (receive audio) is (or can be adjusted to) approximately 500 mV a direct connection to the TNC-220 audio input may be made. If the audio output is significantly greater than 700mV the swamping resistor in the TNC-220 may have to be changed to accomodate the radio.



NOTE: This diagram shows only one port.

Fig. 3-3. Accessory Jack Interface.

If you desire to use your radio for both packet and voice, and if your radio does not have an accessory jack and you don't wish to add a connector to your radio, you should construct a separate external interface box. This box will permit simultaneous connection of your TNC-220 and a microphone. It will either have its own speaker or pass the speaker signal through to a connector so that you can monitor the receive audio while your TNC-220 is hooked up. A schematic of an external interface box is shown in Fig. 3-4.



NOTE: This diagram shows only one port.

Fig. 3-4. External Interface Box.

Regardless of whether you use an accessory jack or an external interface box, you should use shielded wire for all signal-carrying leads. The connector types and pinouts depend on the connector jacks on your radio.

When all required cables have been assembled and the TNC-220 is connected to the radio, you are ready to set the levels according to the following procedure.

NOTE: Use a 500k ohm linear taper variable resistor for R(S) in Fig. 3-3 or Fig. 3-4.

WARNING: IF THE TNC-220 IS OPERATED OUTSIDE THE CABINET FOR EVEN A FEW SECONDS, THE 7805 VOLTAGE REGULATOR MUST BE PROPERLY HEATSINKED TO PREVENT FAILURE!! The 7805 regulator must dissipate up to 3.5 watts of electrical power as heat. This is well within the normal ratings for the device IF properly heatsinked. Normally the 7805 is firmly connected to the TNC-220 cabinet for heat transfer. Use the paper clasp provided for a temporary heatsink. It is suggested that a large screwdriver blade be slipped through the 'handles' of the paper clasp to augment the heat dissipation for periods beyond approximately 5 minutes.

1. Connect your TNC-220 to the radio. Connect the microphone to the radio, or to the interface box if one is being used. Connect the radio to a dummy load. Turn on your TNC-220 and computer and start your terminal program.

Since the TNC-220 interfaces to two radio transceivers, you will need to go through the following procedure separately for each radio. Use the "PORT 2" command to cause the TNC-220 to select the VHF radio port and follow the instructions below. Then use "PORT 1" to select the HF port and repeat the procedure. Refer to the TNC-220 board layout diagram for the location of the trimpots to be adjusted.

2. Type in the command **CAL** and type "K" to cause the TNC-220 to key the selected radio with a steady tone.

3. With the TNC-220 keying the transmitter, adjust the transmit audio level as follows. With a small screwdriver, adjust the trimpot to set the output of your TNC-220 for the port you are using for proper modulation level (typically between 3.0 and 4.5 kHz deviation for Amateur FM). If test equipment is not available, adjust the TNC-220 output as described in steps 2 and 3 of Method 1 above. If using a SSB radio, adjust the trimpot to provide an output reading equal or slightly less than you obtain on CW on that frequency.

4. Type the "K" command to unkey the transmitter, and "Q" to leave the **CAL** mode. You have now set your transmitter deviation to approximately the correct level.

5. The TNC-220 is designed to operate optimally with a 200 to 700 millivolt audio input signal. This is the signal level provided by most modern transceivers at the "Fixed Audio Output" connector on the rear panel. It is recommended that a fixed audio output be used if available. If you use the headphone or external speaker jack on your radio, adjust the audio level to approximately 500 mV with the aid of a high impedance voltmeter or oscilloscope. If no such measuring equipment is available, set the volume control at the position normally used for comfortable listening. This should provide an adequate signal output. NOTE: The DCD LED on the TNC-220 is software controlled and is not suitable for setting audio input levels as is done on the TNC-200.

Chapter 4
BEGINNING OPERATION

This chapter will guide you through the basics of packet radio operation with the TNC-220. Packet radio has a great deal of power and flexibility, and this chapter only scratches the surface of your packet station's capabilities. However, it contains the basic information required to get you on the air.

First Steps

Turn on your TNC-220. You should see the following display.

```
|A

Pac-Comm Packet Radio Systems TNC-220
AX.25 Level 2 Version 2.0
Release n.n.n  mm/dd/yy - nnK RAM
Checksum $xx
cmd:
```

The first five lines are the sign-on message, which you will normally see only when you power up the TNC-220. (See Software Release Notes or Errata for values of nn and xx.) The Command Mode prompt **cmd:** will appear when the TNC-220 is in Command Mode and is ready to accept your instructions.

You may see some anomalies in the appearance of the display, which will be corrected as you proceed through this chapter. The sign-on message may appear double-spaced, or characters you type may be displayed twice. You may even see incorrectly displayed characters.

The next section describes the commands you will use to configure the TNC-220 for proper text display for your particular computer. You may not use these commands again unless you change computers or terminal programs. The following sections, "Basic Operation" and "Monitoring Channel Activity," describe the commands you will use for your everyday packet operations. With these commands you will be ready for the section, "Your First Packet connection." The last section of the chapter, "Special Input Characters" contains information on input editing and other special characters used by the TNC-220.

Entering Commands

In the examples in this chapter, text that you are supposed to type will appear in **boldface**. Text typed by the TNC-220 will appear in normal type:

cmd:**ECHO**

This means that you are supposed to type the text, "ECHO", following the Command Mode prompt which the TNC-220 typed. End the line with a carriage return. All command entries will end with a carriage return, abbreviated <CR>. The <CR> at the end of a command normally won't be shown in the examples.

Serial Port Configuration

This section describes how to set up your TNC-220 to work with your terminal.

Parity and Word Length

If messages from your TNC-220 appear garbled, with incorrectly displayed characters, you may need to change the TNC-220's serial port parity and word length. (We assume that you have set the baud rate correctly in Chapter 2. The most common parity and word length combinations are 7 bits, even parity (the TNC-220 default), and 7 bits, space parity. The TNC-220 default will probably be accepted even if your computer actually uses the latter setting. If your computer receives 8 bits as data, you may have to set space parity, since text may otherwise be interpreted as graphics or other special characters. To set 7 bits, space parity, use the combination immediately below. To return to 7 bits, even parity, set the second combination:

AWLEN 8	(8-bit words)	**AWLEN 7**	(7-bit words)
PARITY 0	(no parity bit)	**PARITY 3**	(even parity)

One of these combinations will satisfy most computers. You are more likely to require a different setting if you have a terminal rather than a computer, or if you have configured your terminal port for some special application. If your computer requires odd parity, set **PARITY 1**. If your computer detects framing errors, try one of the following settings:

For shorter characters,		For longer characters:	
AWLEN 7	(7-bit words)	**AWLEN 8**	(8-bit words)
PARITY 0	(no parity bit)	**PARITY 1** or **PARITY 3**	

A few computers will frequently lose the first characters of a line when several lines are typed in rapid succession, for example, in the sign-on message. You can give the computer more time between lines by setting **NUCR ON** (delay after <CR>), or **NULF ON** (delay after <LF>). The delay is adjusted by **NULLS**, which sets a number of character-times for the delay.

Echos

You may see two characters on your screen for every character you type, for example:

cmd:**EECCHHOO**

Your computer is echoing the characters you type, and the TNC-220 is also echoing them. In this case, set **ECHO OFF** to stop the TNC-220's echos. If you later use your TNC-220 with a different computer, or with a different terminal program, you may see nothing displayed when you type. In that case, set ECHO ON.

New Lines and Line Wrapping

If everything displayed appears to be double-spaced, your computer is adding an extra linefeed (<LF>) whenever it displays a carriage return (<CR>). Set **AUTOLF OFF** to keep the TNC-220 from also adding a <LF>. If you change equipment you may have to set **AUTOLF ON** to restore the TNC-220's automatic linefeeds.

If you make mistakes while entering commands to the TNC-220, you can make corrections. To erase the last character you typed, enter a <BACKSPACE> character. The TNC-220 will throw away the last character you typed (unless you are at the beginning of a line) and try to erase the character from your screen. Input editing will be discussed in more detail in the section, "Special Input Characters."

The screen-width parameter is set by default to 80, the width of many CRT displays. The TNC-220 will send an extra <CR> (or <CR> <LF> if **AUTOLF** is **ON**) when 80 characters have been displayed on a line. If your computer does not automatically break long lines, you will need to set the screen width to the width of your display. For example, for a computer using a TV set for a display, you would set **SCREENLN 40**. If your computer does automatically break long lines, you should set **SCREENLN 0** to disable this feature on the TNC-220. Otherwise, you will get two <CR>s when the line wraps around.

You are now ready to start setting up the parameters you will use. You may be satisfied with most of the defaults for now, but a few parameters will have to be changed!

Type the text, "**MYCALL XXXXXX**", following the command mode prompt. Of course, you should substitute your own call sign for **XXXXXX**. Don't forget the <CR> at the end of the line.

```
cmd:MYCALL XXXXXX
was      NOCALL
cmd:
```

Your call sign will be used by the TNC-220 as its "address." The TNC-220 responds by telling you the previous value of the **MYCALL** parameter, and gives you a new Command Mode prompt. Now try typing just the command by itself:

```
cmd:MYCALL
MYCALL   XXXXXX
```

This verifies that the TNC-220 accepted your call sign. You can see the current value of most parameters by typing the command that sets the parameter followed by just a <CR>.

Most commands can be abbreviated, and the minimum abbreviations for each command are given in the listings in Chapter 6 and on the **COMMAND REFERENCE CARD**. For the sake of clarity, only the full command names are used in this chapter.

Setting the Time-of-Day Clock

The TNC-220 clock is maintained in software by the Z-80 microprocessor. Each time the power is applied to the TNC-220 the time must be re-entered using the **DAYTIME** command.

Basic Operation

You can practice operating the TNC-220 without actually going on the air. The following section will allow your TNC-220 to "talk to itself," allowing you to become familiar with it before transmitting anything.

Move the push-on jumper on header JP-L (see Table 7-1 and Figure 7-1) to the **"LOOP"** position. Then give the command **FULLDUP ON**. These two steps will cause the TNC-220 to enter digital loopback operation. Don't forget to reset JP-L when you finish the exercise.

A Connecting and Disconnecting Exercise

Packet radio connections are started by a connect process, which sets up the "handshaking" between the two stations that insures error-free communications. Connections are terminated by a disconnect process, which leaves both stations free to start new connections. Try the following commands. Your TNC-220 will connect to itself since it is in digital loop-back, and it will receive all packets that it sends.

```
cmd:CONNECT N2WX (use your callsign instead of N2WX)
*** CONNECTED to N2WX
```

The TNC-220 generates packets initiating and confirming the connection. The packets aren't actually converted to audio signals and transmitted over the radio, but they are otherwise just like packets you will be transmitting later on.

The ***** CONNECTED to** message tells you that the connection was successful. You should also notice that the CON LED has lit up and that you do not see a new **cmd:** prompt on the next line. You are now in Converse Mode, ready to send something, Try it. Type your message, ending the line with a <CR>.

Hello, there.
<u>Hello, there.</u>

The <CR> causes your message to be put into a packet, or "packetized," and transmitted. (We explain in the next chapter how you can use a different character to send packets.) The underlined text is a message that the TNC-220 received in a packet and displayed. Whenever you are in Converse Mode anything you type will be assembled into a packet addressed to the station you are talking to and transmitted. If there isn't a connection in progress, the packet will be sent to the address CQ.

In the example above, your TNC-220 entered Converse Mode automatically after the connect took place. You can also command the TNC-220 to move back and forth between Command Mode and Converse Mode. To return to Command Mode, you must enter a special character, Control-C (abbreviated <CTRL-C>), or else send a BREAK signal. "Control" characters are usually entered by holding down a special control key and then typing another key without releasing the control key. If your keyboard doesn't have a key marked **CTRL** or something similar, consult the documentation for your computer or terminal program to see how to enter control characters. A BREAK signal is a special transmission (not an ASCII character) which your computer may be able to produce.

NOTE: If <CTRL-C> will cause your computer to do something to interfere with packet operations, such as halting the terminal program, and you can't send BREAK signals, you will have to change the character that returns you to Command Mode. See the section on "Special Input Characters," below.

Now type a <CTRL-C>. The TNC-220 doesn't echo the <CTRL-C>, but you should immediately see a Command Mode prompt. To return to Converse Mode, enter the command **CONVERS**:

<CTRL-C>
cmd:**CONVERS**
Whatever I type in Converse Mode is transmitted.
Whatever I type in Converse Mode is transmitted.
<CTRL-C>
cmd:

To terminate the connection, you must type the **DISCONNE** command. The TNC-220 will transmit packets terminating the conversation and notify you when the disconnect is complete: (Remember to reset JP-L back to NORM and turn FULLDUP OFF.)

cmd:**DISCONNE**
*** DISCONNECTED

An actual connection might be ended by the other station. In that case, you would see the ***** DISCONNECTED** message without having issued the command.

Digipeating

You may wish to have a connection with another packet station that is beyond your direct radio range. If a third packet station is on the air and both you and the station you want to talk to are in range of this third station, that station can relay your packets. You set up the packet routing when you initiate the connection. Your TNC-220 will then automatically include the routing information in the packets it sends.

The diagram below shows an example situation in which digipeating is useful.

```
                    AD7I
N2WX _________/           \_________ KE8CW
```

You are station N2WX, and you want to have a packet connection with KE8CW. There is a mountain in the way and you are not in simplex range of each other. However there is a station located on the ridge, AD7I, which is in range of both you and KE8CW.

You direct the TNC-220 to set up a connection to KE8CW using AD7I as an intermediate digipeater as follows:

cmd:**CONNECT KE8CW VIA AD7I**

You can specify a routing list of up to eight intermediate stations. For example, consider a modification of the example above:

```
                ___
N2WX _________/    \_________ KE8CW
      .                    .
        K4NTA . . . KV7D
```

AD7I has turned off his station, but you can contact KE8CW by going around the mountain through K4NTA and KV7D. This time you issue the connect command like this:

cmd:**CONNECT KE8CW VIA K4NTA, KV7D**

You specify the digipeaters in the order you would encounter them going from your station to the station to which you wish to connect.

Your station can also act as a digipeater for other stations. This doesn't require any special actions on your part -- your TNC-220 will do everything automatically. If your station is digipeating, you may occasionally notice your transmitter keying during lulls in your own conversations.

Unsuccessful Connections

Sometimes you will initiate a connect sequence that can't be completed. The station may not be on the air, or it may not be within range of your station. You may have even mis-typed the other call sign. If the TNC-220 does not get a response to its first connect packet, it will try again. You can control the number of attempts the TNC-220 will make with the command **RETRY**. The default number of retry attempts is 10. If the TNC-220 doesn't get an answer after this number of transmissions, it will give up and display the message

```
*** retry count exceeded
*** DISCONNECTED
```

The retry count is also used once the connection has started. Each transmission sent to the other station is "acknowledged," or ACKed by the other station, and vice versa. The ACK means that the packet was received and that the CRC checksum indicated that it was received without errors. This is the means by which packet radio can ensure error-free communications. Sometimes a packet won't be received correctly by the other station, either because of accidental interference from another packet station (a collision), or because of other channel noise. If your TNC-220 doesn't get an ACK soon enough, it retransmits the packet and increments the retry count. If the count set by **RETRY** is exceeded, the TNC-220 will automatically disconnect and display the same message:

```
*** retry count exceeded
*** DISCONNECTED
```

The automatic disconnect feature keeps a TNC-220 from indefinitely retransmitting a packet and tying up the channel under hopeless conditions. For example, an intermediate digipeater might have been shut down, or the RF channel might have deteriorated badly. The other operator might have even turned off his station without disconnecting. If you are operating under special conditions, such as a marginal HF channel, you can set **RETRY 0** to disable all automatic disconnects (the retry limit is never reached).

Monitoring Channel Activity

In addition to displaying messages from the station you are connected to, your TNC-220 can allow you to monitor other packet activity on the channel. TNC-220 will also keep track of stations heard during a session. This section describes the monitor functions.

Monitoring is enabled or disabled by the **MONITOR** command. You can try this out in digital loop-back mode while disconnected. Type:

```
cmd:MONITOR ON
cmd:CONVERS
This is a test packet.
N2WX>CQ:This is a test packet.
```

Since you aren't connected to another station your packets are sent to the address "CQ," i.e., anyone. The packet you sent was "heard" by the TNC-220 and displayed, along with the sending station and the destination.

If you also want to see any intermediate digipeater stations being used, you can set **MRPT ON**. This feature would be useful if you later want to connect to one of the stations you are monitoring and will need a digipeater route in order to reach it. For example, you might see the following display:

KF4EF>KF4LG,W4ORA:Hello, Bill!

This packet was sent from KF4EF via W4ORA to KF4LG.

If there are several digipeaters, or if the message lines are long, the display may be difficult to read. You can put the address header on a separate line from the text by setting **HEADERLN ON**:

KF4EF>KF4LG,W4ORA:
Hello, Bill!

Ordinarily, your TNC-220 will stop displaying monitored packets if you connect to another station, permitting you to converse without interruption. If you want to monitor activity while connected to a packet station, set **MCON ON**.

To display a list of stations heard since the last time your TNC-220 was powered up, type:

cmd:**MHEARD**
WB4ZNW
K4NTA*
VE3ILW

The last eighteen packet stations heard by your TNC-220 are displayed. The entry "K4NTA*" means that K4NTA was heard via a digipeater rather than directly. If the time-of-day clock is set, the date and time each station was most recently heard will also be displayed. You can clear the "heard log" with the command **MHCLEAR**.

You can see the settings of the monitor parameters described above by typing **DISPLAY MONITOR**.

Your First Packet Connection

Although there are still a number of features you should be familiar with for comfortable packet operation, you are probably eager to get on the air and try out your TNC-220. Have another packet operator get on the air on VHF to help you get started. Make sure that your friend will be close enough to ensure solid copy, with no FM "popcorn" noise. It's best if you can get an experienced packet operator to help you get started. Try to have both stations in the same room and operate on low power or into dummy loads.

Table 4-1. Jumper settings for modem tone selections and automatic data rate.

Desired Configuration	JP1	JP2	JP3	JP4
Automatic selection: Port 1-HF tones, 300 bauds Port 2-VHF tones, 1200 bauds	1&2	OFF	ON	ON
Manual selection: Both ports 2025/2225 Hz (HF)	2&3	OFF	ON	OFF
Manual selection: Both ports 1200/2200 Hz (VHF)	1&2	OFF	ON	OFF
Manual selection: Both ports 1300/1700 Hz (EXPERIMENTAL)	2&3	ON	OFF	OFF

The TNC-220 comes with port 1 (J1/Radio 1) set to the 300 bauds, 200 Hz shift for HF operation. Port 2 (J2/Radio 2) is configured for 1200 bauds, 1000 Hz shift. The radio baud rate is <u>not</u> related to the computer baud rate. The modem tones and baud rates will be adjusted automatically when you switch between the ports using the **PORT** command.

Turn on your computer, the TNC-220, and your VHF radio and give the **PORT 2** command to select that radio. Be sure you have adjusted your TNC-220 and radio according to one of the methods described in Chapter 3. When the other station transmits, the DCD LED on your TNC-220 should glow steadily for the duration of the transmission. You can work through the remainder of the examples in this chapter while you try out your TNC-220 on the air.

Starting the Connection

You are ready to initiate a connect. For the sake of example, we will continue to use N2WX in place of your call sign, and we will use K4GFG for your friend's call. Make sure you are in Command Mode, and type:

cmd:CONNECT K4GFG

After a moment you should see the message

*** CONNECTED to K4GFG

and you will be in Converse Mode. Your friend will see the message

*** CONNECTED to N2WX

and he will also be in Converse Mode. You have begun your first connection.

If you have trouble connecting, make sure your microphone drive level is set properly, as described in Chapter 3. It may be helpful to have an experienced packet operator listen to your transmissions and monitor with his TNC. You can also try the following procedure. Both you and your friend should set **MONITOR ON**, enter Converse Mode and send some packets. Each station should display packets sent by the other. If only one station is "hearing" properly, you can concentrate on the modulator and transmitter of that station and the demodulator and receiver of the other station. You can try experimenting with the **TXDELAY** timing parameter for the sending TNC-220. Set **TXDELAY 64** for a long delay. If this solves the problem, you can back off to the smallest value that works consistently.

If you monitor the radio transmit indicators and listen to the speaker audio from the two rigs, you will have a better idea of what is happening. You radio will be inactive most of the time, even while you are actually typing. When you get to the end of a line and type a <CR>, your radio will be keyed briefly and your friend will hear a "brrrraaaap" on his speaker. As your message is displayed on his computer, his radio will be keyed for an even shorter time and you will hear a "brraap" on your speaker. This is the ACK, or packet acknowledgment coming back. Your TNC-220 takes note that the packet was received correctly, but nothing is displayed on your screen.

Digipeating

Now that you are on the air, you can try out the TNC-220's digipeating capabilities. This is more realistic if you have at least three stations participating, but you can get the feel for it with two stations.

Return to Command Mode and disconnect from the other station:

<CTRL-C>
cmd:DISCONNE
*** DISCONNECTED

Now issue the following command.

cmd:CONNECT N2WX VIA K4GFG

(Remember to substitute your callsign for N2WX and your friend's call for K4GFG). You are initiating a connect to yourself, as you did above in digital loop-back mode, but you are using an "RF loop-back." You transmit packets to your friend's TNC-220, which relays them back to you. When the connection is established you see

*** CONNECTED to N2WX VIA K4GFG

and you will be in Converse Mode. Your friend won't see anything displayed on his computer and his TNC-220's state won't be affected at all by your connection. In fact, your friend could issue this connect request,

cmd:**CONNECT K4GFG VIA N2WX**

and carry on a separate conversation completely independently. Monitor the radio transmit indicators and listen to the speaker audio. See if you can follow the packets and the acknowledgements back and forth.

Monitoring on the Air

While you and your friend are separately connected, type to try out the TNC-220's monitor functions.

<CTRL-C>
cmd:**MONITOR ON**
cmd:**MCON ON**
cmd:**CONVERS**

You will be able to see both your and your friend's "conversations." Also try **HEADERLN** ON and **MHEARD**.

A convenient feature of the TNC-220 is the ability to be permanently connected to two radios and to have the radio used to be selected by a keyboard command. To determine which radio port is in use:

cmd:**PORT**
PORT 2

To select the alternate port:

cmd:**PORT 1**
was 2

When the **PORT** command is used to select the alternate radio, the baud rate, tones, and shift are selected automatically according to the jumper settings on the TNC-220 circuit board. Parameters such as TxD, FRACK, MAXFRAME, PACLEN, SOFTDCD, etc. must be altered manually. The default settings for these commands have been selected for proper operation on both HF and VHF. Set TxD to the higher value required by either of your radios. See Chapter 5 for tips on optimizing these parameters for the operating conditions in only one type of service.

Special Input Characters

The TNC-220 has a number of special characters that can be used to control its actions. Many of these special characters can be used to "edit" commands and packet text as they are entered. These features can all be customized to suit you and your computer. Most of the special input characters we will describe are active in both Command Mode and Converse Mode; the exceptions will be noted.

The character used to return to Command Mode from Converse Mode is by default a <CTRL-C>. (Sending a BREAK signal also works.) This character does nothing in Command Mode, so if you accidentally enter it twice you won't mess up the next command line. You can change the Command Mode entry character with the command **COMMAND**. This is one of several commands that set special character functions. You can choose any character for this function, by entering the ASCII character code for the key. For example, you can use a <CTRL-E> to enter Command Mode by setting

cmd:**COMMAND 5**
was $03

The TNC-220 displays the previous value in hex, and you can also enter character codes in hex if you prefer. All of the special characters described below can be changed in the same way as **COMMAND**.

We have already mentioned that you can erase mis-typed characters by typing the <BACKSPACE> character. You can change this character with the command **DELETE**. If you set **DELETE ON**, you can erase characters by typing the <DELETE> character; setting **DELETE OFF** returns to using <BACKSPACE>. You will probably want to use the same key that your computer normally uses to rub out characters. <BACKSPACE> is more commonly used than <DELETE> by personal computers. If you aren't sure whether your rubout key produces <DELETE> or <BACKSPACE> characters, you can try both settings of the **DELETE** command and see which works.

When you rub out a mis-typed character, the TNC-220 will attempt to correct the screen display. This will work for most computers as well as display-type terminals. It won't work for hardcopy-type terminals or possibly with a few computers. If your display doesn't look right after you rub out a character, try setting **BKONDEL OFF**. The TNC-220 will not try to correct the display but will indicate the rubout with a "\" character (<BACKSLASH>). You can restore display correction by setting **BKONDEL ON**.

If you make several mistakes in a line, or if you change your mind, you may want to cancel the whole line rather than rubbing out the characters one at a time. You can cancel the line by typing <CTRL-X>. The TNC-200 will display a <BACKSLASH> followed by <CR>. If you are in Command Mode, you will see a new prompt:

```
cmd:Hi, John, how are you?<CTRL-X>\
[You started typing text while in Command Mode.]
cmd:CONVERSE
Hi, John, how are you?
```

The cancel-line character can be changed to any ASCII character by the command **CANLINE**.

If you have changed your input by rubbing out and retyping characters, you may want to see a "fresh" copy of your input, especially if you have set **BKONDEL OFF**. The TNC-220 will retype the line you are entering when you type <CTRL-R>:

```
cmd:CONNECT KB7\\\WA7<CTRL-R>\
[You mis-typed the call sign.]
cmd:CONNECT WA7GXD
```

Here the user mis-typed the first three characters of the call sign and rubbed them out. The TNC-220 displayed "\" for each character rubbed out. The user then retyped the characters correctly and redisplayed the line. He finished typing the call sign on the new line. The redisplay-line character can be changed to any ASCII character by the command **REDISPLA**.

If your TNC-220 displays information faster than you can read it before it scrolls off the screen, you can halt the display by typing <CTRL-S>. To resume output from the TNC-220 to your computer, enter <CTRL-Q>. These characters can be changed to any ASCII character by the commands **STOP** and **START**, respectively.

You may occasionally want to include one of the special input characters in a packet. For example, to send several lines at once in the same packet, you would have to include <CR> in the packet at the end of each line, bypassing its "send-packet" function except at the actual end of the packet. You can include any character in a packet including all special characters by prefixing it with the pass character, <CTRL-V>. For example,

```
I wasn't at the meeting.<CTRL-V><CR>
What happened?
```

Ordinarily, this message would be sent as two packets. By prefixing the first <CR> with <CTRL-V>, the operator sends it all at once, but maintains the <CR> in the text. The pass character can be changed to any ASCII character by the command **PASS**.

If you intend to use your packet station for "advanced" applications, such as a computer Bulletin Board or binary file transfers, you should continue with Chapter 5, Operation II: Further Details. However, Chapter 4 contains all the information that is needed for most packet operations. For a full description of all commands, refer to Chapter 6, Commands.

Chapter 5
OPERATIONS II: ADVANCED OPERATING

This chapter describes some aspects of packet operation for special applications such as a computer Bulletin Board, binary file transfers, and multi-connections. You will also find this material useful if you intend to operate packet through satellites or if your radio has special timing or other requirements.

We will use the term "computer" to refer to computers or terminals. In the command examples, the TNC-220's prompts and other messages are shown in ordinary type, your responses are shown in **bold face**, and received packets are shown underlined. Commands and other special keywords are shown in upper case; other text entered to the TNC-220 is shown in upper and lower case.

Special Characters

The TNC-220 recognizes a number of special characters for input editing, flow control, and other control functions. You can change any of these special characters to customize your TNC-220 to suit your applications, your computer, or your whim. Most of the characters are set by commands which specify the ASCII character code for the desired character. You can disable any special character feature by setting the character value to 0. Input editing characters may be disabled with no serious effects. You should use caution in disabling the flow-control or Command Mode entry characters. Also be careful not to set two special characters to the same value.

Special characters are normally set to various control characters. Control characters are entered by holding down a special control key while typing another key. For example, control-C, or <CTRL-C> is entered by holding down the control key while typing **C**. If your computer doesn't have a special control key, you will have to consult your computer's documentation to see how to enter these characters. If you will have difficulty entering control characters, you can change the special characters to, for example, seldom-used punctuation.

The action of each special character is described in detail under the entry in Chapter 6 for the command that sets that character.

You can enter the code for a character in either hex (base 16) or decimal notation. The TNC-220 displays character codes in hex. A number in hex notation is indicated by beginning the number with a **$**. The "digits" of a hex number represent multiples of powers of 16. The values 10 through 15 are represented by the letters **A** through **F**, which may be upper or lower case. For example: $1B = 1 x 16 + 11 = 27.

Tables of ASCII character codes are available in most computer manuals. A table of ASCII codes for control characters appears below.

Operating Modes

The TNC-220 has three operating modes. Two of these modes, Command Mode and Converse Mode, were discussed in Chapter 4. The third mode, Transparent Mode, is a data-transfer mode like Converse Mode but is intended primarily for computer data interchange rather than human conversation. We describe all three of these modes below.

Command Mode

Command Mode is used to enter commands which alter the TNC-220's operating parameters. The other modes are entered from Command Mode. When the TNC-220 is in Command Mode, the Command Mode prompt, "cmd:" is printed at the beginning of each input line. Note, however, that if the TNC-220 has received and displayed packets, the prompt may have scrolled off the screen.

The TNC-220 will be in Command Mode after a reset or power-up. After a power-off, power-on sequence, all operating parameters of the TNC-220 are re-initialized to the parameter stored in battery backed-up RAM (bbRAM) by the resident software. The values of most parameters are stored in a permanent but easily changed form in the bbRAM memory. After the **RESTART** command is issued all operating parameters are reset to the default values stored in EPROM. The TNC-220 has also re-initialized itself exactly as it would upon power-up, and typed a sign-on message. You probably won't use this command often. The TNC-220 will automatically reload the bbRAM at power up if it finds that the data is bad.

Table 5-1. ASCII Codes for Control Characters

Dec	Hex	Control	Mnemonic	Dec	Hex	Control	Mnemonic
0	$00	<CTRL-@>	NUL	16	$10	<CTRL-P>	DLE
1	$01	<CTRL-A>	SOH	17	$11	<CTRL-Q>	DC1
2	$02	<CTRL-B>	STX	18	$12	<CTRL-R>	DC2
3	$03	<CTRL-C>	ETX	19	$13	<CTRL-S>	DC3
4	$04	<CTRL-D>	EOT	20	$14	<CTRL-T>	DC4
5	$05	<CTRL-E>	ENQ	21	$15	<CTRL-U>	NAK
6	$06	<CTRL-F>	ACK	22	$16	<CTRL-V>	SYN
7	$07	<CTRL-G>	BEL	23	$17	<CTRL-W>	ETB
8	$08	<CTRL-H>	BS	24	$18	<CTRL-X>	CAN
9	$09	<CTRL-I>	HT	25	$19	<CTRL-Y>	EM
10	$0A	<CTRL-J>	LF	26	$1A	<CTRL-Z>	SUB
11	$0B	<CTRL-K>	VT	27	$1B	<CTRL-[>	ESC
12	$0C	<CTRL-L>	FF	28	$1C	<CTRL-\>	FS
13	$0D	<CTRL-M>	CR	29	$1D	<CTRL-]>	GS
14	$0E	<CTRL-N>	SO	30	$1E	<CTRL-^>	RS
15	$0F	<CTRL-O>	SI	31	$1F	<CTRL-_>	US
				127	$7F	<DELETE>	DEL

The following commands set special characters which are active in Command Mode. Refer to the discussions of these commands in Chapter 6 for details on the operation of the characters in Command Mode. Also see the special input character section in Chapter 4.

CANLINE	Cancel current line
CANPAC	Cancel output (Command Mode function only)
DELETE	Character deletion
PASS	Insert following special character
REDISPLA	Re-display current line
START, STOP	User's flow control characters (sent to TNC-220)
XOFF, XON	TNC-220 flow control characters (sent to terminal)

The following commands enable display features which are active in Command Mode. Refer to the discussions of these commands in Chapter 6 for details on the operation of these characters in Command Mode. Also see the section on terminal configuration in Chapter 4.

AUTOLF	Add <LF> after <CR> in data sent to terminal
BKONDEL	Echo after character deletion
ECHO	Automatic echo of serial port input
FLOW	Type-in flow control
LCOK	Lower case translation
LFIGNORE	Line Feed characters ignored
NUCR	Nulls after <CR>
NULF	Nulls after <LF>
NULLS	Null count
SCREENLN	Automatic <CR> insertion

Entering Data-Transfer Modes

There are several ways to enter a data-transfer mode from Command Mode. You can type the command **CONVERS** (or the single-letter command "K") to enter Converse Mode or the command **TRANS** to enter Transparent Mode, and the TNC-220 will immediately enter the specified mode. The TNC-220 will automatically enter a data-transfer mode if you are in Command Mode when a connection is completed. You can specify the data-transfer mode for automatic entry with the command **CONMODE**:

cmd:CONMODE TRANS

will specify Transparent Mode, and

cmd:CONMODE CONVERS

will return to the default choice of Converse Mode.

The timing of the automatic entry into data-transfer mode depends on whether you or the other station initiated the connection. If you receive a connect request which your TNC-220 accepts, you will enter data-transfer mode when the TNC-220 sends the connect acknowledgment (ACK) and types the message

*** CONNECTED TO <callsign>.

If you initiate the connection with the **CONNECT** command, you can control the timing of the mode change with the command **NEWMODE**. If **NEWMODE** is **OFF**, the mode will change when the connect ACK is received and the ***** CONNECTED TO: <callsign>** message is typed. If **NEWMODE** is **ON**, you will enter data-transfer mode immediately, without waiting for a successful connection. Any text sent to the TNC-220 at this point will be queued up in packets which will wait for a successful connection before being sent. If the connect attempt fails, you will be returned to Command Mode. You will also be returned automatically to Command Mode when either station disconnects and ends the QSO.

Converse Mode

The data mode used most often for ordinary QSOs is Converse Mode. In Converse Mode, the send-packet character causes the input to be packetized for transmission. If you type a full packet-length of characters without typing the send-packet character, your input will be packetized and transmitted anyway.

The default send-packet character is <CR>, but you can specify any character with the command **SENDPAC**. You may choose to have the send-packet character transmitted in the packet. If the send-packet character is <CR> it is natural to include it in the packet as part of the text as well as interpreting it as a command. This is accomplished by setting **CR ON**. If you use another character to force packet transmission, you may set **CR OFF** and inhibit transmission of the send-packet character. If you set the send-packet character to something other than <CR>, you can cancel packets of more than one line with the cancel-packet character, which is set with the command **CANPAC**. Single-line packets can be canceled with either the cancel-line character or the cancel-packet character.

To return to Command Mode from Converse Mode you must type the Command Mode entry character, or send a BREAK signal over the serial port. A BREAK is not a regular ASCII character, but it can frequently be transmitted by typing a special key on the keyboard.

A BREAK signal is a continuous mark (or 1) signal on the serial port Transmit Data line lasting approximately 0.2 second. In fact, the timing of the signal is not very important, and most serial ports will recognize a BREAK if the mark signal lasts significantly longer than the time required for a character trans mission. Because of the simple nature of this signal, it is easily possible to generate a BREAK with circuitry external to the computer, thus guaranteeing entry to Command Mode in automatic station operation.

The following commands set special characters which are active in Converse Mode. Refer to the discussions of these commands for details on the operation of the characters in Converse Mode.

CANLINE	cancel current line
CANPAC	cancel current packet
COMMAND	Command Mode entry
DELETE	character deletion
MFILTER	characters to be filtered in monitored packets
PASS	insert following special character
REDISPLA	re-display current line
SENDPAC	send current packet
START, STOP	user's flow control characters (sent to TNC-220)
XOFF, XON	TNC-220 flow control characters (sent to terminal)

The following commands enable display features which are active in Converse Mode. Refer to the discussions of these commands for details on the operation of these characters in Converse Mode.

8BITCONV	Retain high-order bit from serial port in converse mode
AUTOLF	Add <LF> after <CR>
BKONDEL	Echo after character deletion
ECHO	Automatic echo of serial input
ESCAPE	<ESCAPE> translation
FLOW	Type-in flow control
LCOK	Lower case translation
LFIGNORE	Line Feed characters ignored
NUCR	Null characters after <CR>
NULF	Null characters after <LF>
NULLS	Null count
SCREENLN	Automatic <CR> insertion

Transparent Mode

Packet radio is very well suited to transfer of data between computers. In some cases Converse Mode will work well for computer data transfer. However, files such as a .BAT file on a MS/DOS system, a BASIC program, or even a word-processor text file, may contain characters which conflict with special characters in Converse Mode. Some of these files may utilize all eight bits of each byte rather than the seven bits required by ASCII codes. If you transfer such files you will have to use Transparent Mode.

Transparent Mode is a data-transfer mode like Converse Mode. In this mode there are no special characters -- everything you type (or everything your computer sends to the TNC-220) is sent over the radio exactly as it was received by the TNC-220. There are no input editing features and there is no send-packet character. Packets are sent at regular time intervals or when a full packet of information is ready. The time interval at which data is packetized is set by the **PACTIME** command.

The display characteristics of the TNC-220 are also modified in Transparent Mode. Data is sent to the computer exactly as it is received over the radio, including all 8 bits of each byte received. Features such as auto-linefeed insertion and screen wrap are disabled, and echoing of input characters is disabled. The parameters that control these features in Command Mode and Converse Mode are not changed by entering Transparent Mode, and all display features are re-enabled when the TNC-220 is returned to Command Mode. Most of the link status messages that appear as the TNC-220 moves between disconnected and connected states are also disabled in Transparent Mode.

In order to permit the Command Mode entry character to be transmitted freely in Transparent Mode, the escape to Command Mode from Transparent Mode has been made a little more complicated. You can still return to Command mode by transmitting a BREAK signal, just as in Converse Mode. You can also utilize the Command Mode entry character in the following way.

You must wait for a time period after typing the last character to be sent. This time is set by the command **CMDTIME**. Following this wait, you must type three Command Mode entry characters (default <CTRL-C>) within an interval **CMDTIME** of each other. After a final **CMDTIME** interval in which no characters are typed, you will see the "cmd:" prompt.

If any characters are typed during this interval (even Command Mode entry characters) the escape will be aborted and all the Command Mode entry characters that have been typed will be sent as packet data. If you set **CMDTIME** to zero you will not be able to escape from Transparent Mode using this second procedure.

Flow Control

Whenever data is transferred to computers (home computers or TNCs), there is a chance that the data will be received faster than the computer can handle it. Some programs try to deal with this by providing data buffers for storing incoming data until the program is ready for it. However, this merely postpones the problem, since there is a limited amount of room in any buffer. In order to prevent loss of data the computer must be able to make whatever is sending data stop sending, and later tell it to resume sending. If you are a home computer user, you are probably already familiar with one type of flow control, which allows you to stop the output from the computer while you read it and restart it when you are ready for more.

The TNC-220's input buffer may fill up in Command Mode if you try to type too long a command. In Converse Mode the buffer may fill up for any of several reasons: you may be using a faster serial port baud rate than the radio data rate; radio data transmission may have slowed down because of noise or other users on the channel; the person or computer at the other end may have stopped output from that TNC. The TNC-220 will signal the computer to stop sending data when there is room remaining for about 80 characters in the buffer. When the buffer fills up entirely, data will be lost. When the buffer empties so that there is room for at least 270 characters, the TNC-220 will signal the computer to start sending data again.

A computer file transfer program may be unable to process data fast enough to keep up with output from the TNC-220. In order to be sure of reading every character, a computer must respond to interrupts from its I/O devices. Some simple programs may poll the input register for new data. If the polling is not done often enough, data may be lost. Some computers disable interrupts during disk accesses. If the program enters a routine which will not allow it to check for data or respond to it, it should signal the TNC-220 to stop sending data.

There are two methods of providing flow control which are supported by the TNC-220. XON/XOFF flow control, sometimes called "software flow control," is accomplished by sending a special character (usually <CTRL-S>) to request that the output stop and another special character (usually <CTRL-Q>) to restart output. Hardware flow control may be used if both computers use the Clear To Send (CTS) and Data Terminal Ready (DTR) lines of the RS-232C standard.

Some commonly used terminal programs and file transfer programs for home computers do not implement flow control in software, and many serial ports do not support hardware flow control. Although the DTR and CTS lines appear at the connector, they may not be used on some computers unless the software reads the state of the CTS line. If you find that the TNC-220 seems to lose data during file transfers, you should immediately suspect a flow control problem.

XON/XOFF Flow Control

If you are using a terminal (rather than a computer) or if your computer does not support DTR/CTS flow control, you should use XON/XOFF flow control, which is enabled by setting **XFLOW ON**. The special flow control characters are set to <CTRL-S> and <CTRL-Q> by default, but they may be changed. The commands **XON** and **XOFF** set the characters which will be sent to the terminal by the TNC-220, and the commands **START** and **STOP** set the characters to be sent to the TNC-220 by the terminal. Your computer may receive as many as 4 characters from the TNC-220 after sending a STOP character, since some characters may already be "enroute" through serial I/O chips.

If you send a STOP (START) character to the TNC-220 when it is already stopped (started), the character will be ignored. If the STOP and START character are the same character, this character will "toggle" the output, turning it off if it is on, and on if it is off.

You can disable XON/XOFF flow control in one direction only by setting the appropriate flow control characters to 0. If you do this, the TNC-220 will automatically use CTS flow control to stop input from the terminal. XON/XOFF flow control is normally disabled in Transparent Mode, since all characters are treated as data. If you cannot use DTR/CTS flow control, you may enable the XON and XOFF characters (the commands from the TNC-220 to the terminal) by setting **TXFLOW ON** and **XFLOW ON**. The START and STOP characters (the commands to the TNC-200 from the terminal) can be enabled in Transparent Mode by setting **TRFLOW ON**. Note that the mode is no longer truly transparent when these features are enabled.

Hardware Flow Control

Hardware flow control is less likely to depend on the programming of a particular communications program. DTR and CTS are normally used for flow control signals in Transparent Mode. The command **XFLOW OFF** enables hardware flow control in Converse Mode and Command Mode. Your computer may receive as many as 2 characters after it signals the TNC-220 to stop sending, since some characters may already be "enroute" through serial I/O chips. Refer to Chapter 7, Hardware for details on the interface required for hardware flow control.

Type-in Flow Control

Type-in flow control, enabled with the command **FLOW**, is really a display feature. It can keep the TNC-220 from interrupting you with incoming packets when you are in the middle of typing a command line or an outgoing packet. As soon as you type the first character of a line, the TNC-220 will put a "hold" on all output (except for echoing your input). The "hold" remains in effect until you type a <CR> to end the command line, or a send-packet character to mark the end of a packet, or until you erase or re-display the line you have started.

Some computers have difficulty simultaneously sending and receiving characters over the serial port. This is most commonly the case for computers with "software UARTs." Type-in flow control will improve the operation of such computers with the TNC-220.

Additional Packet Operations

Station Identification

Your station identification (call sign) is set with the command **MYCALL**, as described in the previous chapter. If you will have more than one station on the air operating with the same call sign, they must be distinguished -- no two stations can have identical station identifications, or the packet protocol will fail. You can distinguish additional stations by setting the "secondary station ID", or SSID. This is a number from 0 to 15, appended to the call sign with a dash:

cmd:**MYCALL W3IWI-3**

If you don't specify the SSID extension, it will be 0, and the TNC-220 won't explicitly show SSIDs that are 0. If you want to connect to a station with a SSID other than 0, or use such a station as a digipeater, you must specify the SSID:

cmd:CONNECT N4KPU-2
or
cmd:CONNECT **WB9FLW** VIA N7CL-5

The TNC-220 can send an automatic identification packet every 9-1/2 minutes when your station is operating as a digipeater. You can enable this with the command **HID ON**. An ID packet is displayed as follows by a monitoring station:

K4RHD-3>ID:K4RHD/R

You can request a final identification as you take your station off the air with the command **ID**. The TNC-220 will only send identification packets if it has been digipeating.

Automatic Operations

Normally, any packet station can be used by other stations for relaying, or digipeating, packets to a more remote destination. If you don't want your station digipeating packets, you can give the command **DIGIPEAT OFF**. Unless there are special circumstances, such as a station operating on emergency power, most packet operators set **DIGIPEAT ON** in the spirit of amateur cooperation.

Your station will normally accept a connect request from another station if it isn't already connected. You can disable this capability by setting **CONOK OFF**. If you receive a connect request when **CONOK** is **OFF**, the TNC-220 will display the message

```
*** connect request: <callsign>
```

and send a "busy signal" rejection packet to the other station. If you receive a rejection packet from a station you try to connect to, your TNC-220 will display

```
*** <callsign> busy
*** DISCONNECTED
```

If you want to have a special message sent automatically to stations connecting to you, you can specify the message with the command **CTEXT**. This message can consist of any text string up to 120 characters, and you may include <CR>s by prefixing them with the pass character:

```
cmd:CTEXT Sorry, I can't talk right now.<CTRL-V><CR>
I'll be on the air again after 8 PM.<CTRL-V><CR>
Joe
```

In order for this message to be sent to stations connecting to you, you must set **CONOK ON** so that the connection takes place (default), and enable the automatic message with **CMSG ON**. You can cause your TNC-220 to issue a disconnect request as soon as the connect message is acknowledged by issuing the command **CMSGDISC ON**. NOTE: Be sure to turn **CMSGDISC** off when you resume in-person operation of your station to prevent premature disconnects.

If you want to leave you station on but inhibit transmitting, you can set **XMITOK OFF**. If you do this, you would normally set **CONOK OFF** as well.

You can have your station periodically send an automatic message by enabling "beacons." A beacon can be used to make general interest announcements, provide packets for other stations to use to test their ability to receive, or announce the presence of a bulletin-board operation. The beacon message is set with the command **BTEXT**, which works the same way as the **CTEXT** command. You enable beacon transmission and set the frequency at which beacons are sent with the command **BEACON**. To transmit the beacon at 5 minute intervals, for example, give the command

```
cmd:BEACON EVERY 30
```

The beacon function also has a transmit-after mode, enabled by using the keyword **AFTER** in place of **EVERY**, in which a beacon packet is only transmitted after activity is heard on the channel. This feature might be used to leave an announcement for other packet users. If someone transmits on an otherwise idle channel, a beacon can be sent a short time later. No beacons are sent in this mode if there is a lot of packet activity on the channel, since the required period of quiet will not occur.

Packet Formatting

The maximum length of a packet is determined by the command **PACLEN**. If you type more than the maximum number of characters without entering a send-packet character, the TNC-220 will transmit a maximum-length packet. In Transparent Mode, a packet will be sent if the maximum number of characters is entered before the delay conditions set by **PACTIME** force a packet to be sent. Some TNCs may not be able to accept packets longer than 128 characters.

If you have set the send-packet character to <CR>, you probably want the <CR> to be included in the packet for display at the other end. If you set the send-packet character to a special non-printing character, you probably want the character to be treated as a command only. The command **CR** controls whether the send-packet character is to be echoed and included in the packet.

You can add a <LF> after each <CR> included in your packets by setting **LFADD** ON. If the other station reports that lines are overprinted on his display, and he can't remedy the situation at his end, you can enable this function.

Commands Affecting Protocol

This section describes some of the commands that affect the operation of the packet protocol. Details of the protocol are given in Chapter 9.

The TNC-220 implements AX.25 Level 2 protocol, a set of rules for formatting messages to other TNCs. The TNC-220 uses version 2 of AX.25 Level 2 protocol by default. This may be set to version 1 with the command **AX25L2V2 OFF**. Digipeating may not be successful if some TNCs are running Version 1.0 and some are running Version 2.0. In addition, the command **CHECK** controls a timing function that depends on the protocol version selected.

You can specify the "address" to be used for unconnected packets, as well as intermediate digipeaters with the **UNPROTO** command. The format is similar to that of the **CONNECT** command:

cmd:**UNPROTO** QST VIA WU4W

The default address for unconnected packets is **CQ**.

The following functions may be useful for tracking down protocol problems. They are seldom useful for ordinary packet operations. The error-checking function of the protocol is disabled for monitored packets with the command **PASSALL**. If you set **PASSALL ON**, any "packet" will be displayed if it meets the following conditions: It must start with a flag sequence; and it must contain an integral number of 8-bit bytes. The **TRACE** command enables the display of the address and control fields of packets, as well as the text. The trace function displays all bytes in hex as well as ASCII equivalents.

Packet Timing Functions

Transmit Timing

Radio equipment varies greatly in the time delays required for switching from receive to transmit and from transmit to receive. If the TNC-220 starts sending data before the transmitter is operating or before the receiver has had time to switch from transmitting and lock up on the incoming signal, the packet will not be received properly. The delay between transmitter keyup and the beginning of data transmission is controlled by the command **TXDELAY**. During the time the TNC-220 is keying the transmitter but not sending data, it will transmit a synchronizing signal (flags).

If you are transmitting packets through an audio repeater, you may require a considerably greater keyup delay than is required for direct communications. Furthermore, the extra keyup delay is not required if the repeater has not had time to "drop" since the last transmission. The command **AXDELAY** allows you to specify an additional keyup delay to allow the repeater receiver and transmitter to lock up. The command **AXHANG** sets the time the TNC-220 will assume is required for the repeater to drop. If the TNC-220 has detected channel activity recently enough that the repeater transmitter should still be on, it will wait only the **TXDELAY** time before sending data, rather than adding an **AXDELAY** time as well. The commands **TXDELAY**, **AXDELAY**, and **AXHANG** all set times in units of 10 ms. If **AXDELAY** is in effect the total keyup delay will be

Keyup delay = (TXDELAY + AXDELAY) * 10

in milliseconds. If channel activity has been heard more recently than AXHANG*10 ms ago,

Monitor Functions

The TNC-220's protocol is designed for setting up "circuits" between two stations. However it can also operate in a mode more suitable for a "net" or "round-table" discussion with several participants, although reliable reception of all transmissions by every station cannot be guaranteed. This is done by enabling the monitor functions. Most of the monitor functions are described in Chapter 4.

Monitoring is enabled by the command **MONITOR ON**, and separate monitor functions are individually enabled.

If connected packet QSOs are taking place on the frequency of your group conversation, you may wish to ignore all connected packets while your group operates in unconnected mode. The command **MALL OFF** causes the TNC-220 to ignore connected packets.

If you want to be able to monitor packet activity when your station is not connected, but have all monitoring automatically cease when you connect to someone, set **MCON OFF**.

To monitor stations selectively, you can set up a list of up to eight callsigns with the command **LCALLS**. The callsigns in this list are regarded as "buddies," i.e., the only stations you want to listen to if **BUDLIST** is **ON**. Otherwise, the stations in the list will be ignored, and all other stations will be monitored.

You can operate a group conversation with some data integrity by having the stations connect in pairs and setting **MALL ON** and **MCON ON**. This does not insure that every packet is received at every station, but it does insure that a packet involved in a collision will be retried. If you have an odd number of stations participating in this sort of conversation, one station can connect to himself via another station as digipeater.

For example, K5OXE, WS4Z, W4MUA, WB4ZNW and W2EB wish to carry on a group conversation. In order to make all the transmissions as reliable as possible, the following connections are made.

K5OXE connects to WS4Z
W4MUA connects to WB4ZNW
W2EB connects to W2EB via WB4ZNW

If each station specifies **MCON ON** and **MALL ON**, each station will see the packets sent by all the others.

Ordinarily, only text packets are displayed. If you desire to monitor all packet activity on frequency including all of the 'supervisory' frames that contain no user data, set **MCOM ON**. This is especially useful during HF operation when many packets are heard, but few are being displayed. This allows every correctly received frame to be displayed and speeds up the tuning process. For example:

```
WA7GXD>KV7D <I C S0 R0>: Hi Dan,

WA7GXD>KV7D <I C P S1 R0>: have you been on HF lately?

KV7D>WA7GXD <RR R F R2>
KV7D>WA7GXD <I C P S1 R2>: I was just thinking about that.

WA7GXD>KV7D <RR R F R2>
WB2SPE>KV7D <C>
KV7D>WB2SPE <DM>
KV7D>WA7GXD <I C P S2 R2): Good conditions now...

WA7GXD>KV7D <RR R F R3>
WA7GXD>KV7D <I C P S2 R3>: Lets QSY to 14107.
```

When MCOM is turned ON, the interpertation of the command byte in each monitored frame is also displayed.

The basic structure of the MCOM information is-

< cmd_type ((C|R) (P|F)) [Sn] [Rn] >

The "<" and ">" characters are always present, and serve to delimit the new MCOM information. Cmd_type may take any of the following values:

C	-	SABM connect
D	-	DISC disconnect
UA	-	UA unnumbered acknowledge
DM	-	DM disconnected mode
FRMR	-	FRMR Frame reject
UI	-	Unnumbered information, a.k.a. Beacons, ID frames, etc.
RR	-	explicit acknowledgement, has R field
RNR	-	receiver not ready, " "
REJ	-	reject, " "
I	-	Sequenced information packets, has both S and R field

C|R, P|F - The C(ommand), R(esponse), P(oll) and F(inal) frame types are only used in AX.25 level 2 version 2.0 mode. These types are not displayed when packets not using version 2.0 are monitored.

Sn - Sn is displayed for sequenced information (I) frames. ´n´ is the frame´s sequence number and is an integer 0..7.

Rn - Rn is present in both I frames and RR-RNR-REJ frames. The ´n´ value monitored shows the sequence number that the sending station expects it´s peer will use for the next new sequenced information frame.

You can cause the TNC-220 to "filter" certain characters from monitored packets with the command **MFILTER**. This allows you to remove, for example, form-feeds, bell characters, or clear-screen characters that may be necessary to the stations involved in a connection, but which may interfere with your display. You can specify up to four characters by giving the ASCII character codes in hex or decimal. Use the command **LFIGNORE** to filter out line feed characters.

"Health" Features

The CON and STA LEDS can be made to indicate the condition of the processor on the TNC-220 in place of their normal functions. The command **HEALLED ON** will cause the two LEDS to turn on and off in a random pattern if the processor is properly executing the software. **HEALLED OFF** returns the CON and STA LEDS to their normal functions.

Sixteen counters, all of them 16 bits wide, are provided to monitor a number of parameters in the TNC-220. They are ALWAYS initialized to 0000 on power up or "RESTART". The counters and the setting of HEALLED (HEALTH LED) are displayed in response to the command DISPLAY HEALTH.

ASYRXOVR: Increases when the software does not service the asynchronous receiver in time. Indicates data from the user to the TNC is being dropped. This error counter should never become non-zero under supported data rates.

BBFAILED n: Counts number of times bbRAM checksum was in error.

DIGISENT: Each frame digipeated by this TNC causes the counter to increase.

HOVRERR: Increases when HDLC receiver is not serviced rapidly enough and data is lost. This counter should never increment at any supported data rate.

HUNDRERR: Increases when the HDLC transmitter is not serviced rapidly enough and frames are aborted. This counter should never be non-zero at any supported data rate.

RCVDFRMR: Increases when Frame reject frames are received from a connected station.

RCVDIFRA: Increases for each reception of an I-frame from a connectee.

RCVDREJ: Increases for each reception of an REJect frame from a connectee.

RCVDSABM: Each received SABM frame addressed to the TNC causes this counter to be increased by one.

RXCOUNT: Increases when any frame is received with good CRC (or any CRC if PASSALL is turned on).

RXERRORS: Increments each time a received frame is thrown out due to it being too short, suffering overrun(s), or it having a bad CRC. Latter occurs only when CRC checking is enabled (i.e. PASSALL is OFF). This counter will often increment in the presence of noise.

SENTFRMR: Increments each time a Frame reject frame is transmitted.

SENTIFRA: Increases by one each time an I frame is sent.

SENTREJ: Whenever a REJect frame is transmitted, this counter is incremented.

TXCOUNT: Incremented whenever a frame is correctly transmitted.

TXQOVFLW n: Counts how many times frames were discarded because the outgoing frame queue was too small.

Real-Time Clock and Time Stamping

You can enable the TNC-220's real-time clock by setting the date and time with the command **DAYTIME**. Once you have set the clock, you request the time by entering **DAYTIME** with no parameters. The format of date and time display is controlled by the command **DAYUSA**. If the TNC-220 is powered off, **DAYTIME** will be lost.

Monitored packets can be time-stamped if **DAYTIME** has been set. To enable this function, set **MSTAMP** ON. You can also time-stamp connect and disconnect messages with the command **CONSTAMP ON**.

HF and OSCAR

The Pac-Comm TNC-220 is configured for operation on both HF and VHF. Port 1 is configured for best response at to HF tones and port 2 is optimized for 1200 baud tones. The settings of **MAXFRAME** and **PACLEN** have been set to provide satisfactory operation on both HF and VHF.

The requirements for optimum performance with a typical HF or OSCAR 10 path are very different. Lower signal to noise ratios require lower baud rates, noise spikes and fades require shorter packet lengths, and a higher rate of false carrier detects lowers the total usable dynamic range in the audio input. The TNC-220 software can be reconfigured to improve throughput in these environments. For HF operation at 300 bauds, we recommend setting **MAXFRAME** to 1 and **PACLEN** to 40.

The TNC-220 detects a busy channel in one of two ways. If SOFTDCD is ON (default) the TNC-220 monitors the incoming audio signal for the presence of a flag which marks the beginning (or end) of a frame. The presence of a packet signal is indicated by the Data Carrier Detect (DCD) LED. Each time a flag is detected, the DCD is turned on by software and the TNC-220 will start a timing interval which must elapse before the channel is considered to be available.

If SOFTDCD is OFF, the carrier detection and operation of the DCD LED is based on the state of the 8530 DCDA input, pin 19. This pin is jumperable to take its signal from either the on-board AM7910 modem DCD output or the DCD signal from the optional tuning indicator. The on-board modem DCD source will be satisfactory only on a quiet frequency.

On a noisy channel many false carrier detect sequences will occur. For OSCAR and other specialized modes of operation you should set **DWAIT** to 0. The random wait before retry transmissions can be disabled by setting **TXDELAY** 0 and using **AXDELAY** to set the required keyup delay. Of course, **AXHANG** should be 0 for this application. These procedures should not be used on a shared HF channel to avoid interfering with other station's transmissions.

If you are operating a full-duplex radio station (simultaneous transmit and receive), you should set **FULLDUP ON**. The TNC-220 is always electrically capable of full duplex operation, but this parameter causes the protocol to behave differently in acknowledging packets. In addition, the TNC-220 will ignore carrier detect status.

The on-board modem can be completely bypassed at connector J5. You can supply an external modem which uses other modulation methods or higher baud rates. The interfaces available on J4 are TTL levels. Refer to Chapter 7 for more information.

Modem and Port Configuration

The Pac-Comm TNC-220 contains a versatile integrated circuit modem that requires no user calibration. The TNC-220 automatically supports the conventionally used 300 baud and 1200 baud operation of the radio link using the on-board modem, and will operate the modem at 600 bauds through manual jumper settings and use of the **BAUD** command. The software supports 300, 600, 1200, 2400, 4800 and 9600 bauds for use with an external modem.

Switching between radio ports on the TNC-220 is accomplished by the **PORT** command. The default value for **PORT** is 2, the 'VHF' port. The TNC-220 is a single modem device, therefore only one port may be in use at one time. Be sure not to switch ports unless the link state is disconnected and no one is using your station as a digipeater. Whatever link activities exist (such as a connection with another station, beacon message and interval, etc) will be carried over to the other port. It is not possible to use the multiple connection feature of the software across the two ports.

Most 1200 baud operation uses 1000 Hz shift and tones of 1200 Hz and 2200 Hz. These are the default tones for which the TNC-220 modem is optimized on port 2.

Most HF operation below 28 MHz uses 300 baud, 200 Hz shift FSK. Several different tone pairs are in use by various manufacturers. The demodulation center frequency used by the TAPR 1 TNC was 1700 Hz, and HF packet frequencies are conventionally related to that tone pair as tuned on a SSB radio using lower sideband. The TNC-220 modem uses frequencies of 2025 and 2225 Hz for improved noise filtering which produces a center frequency of 2125 Hz. This center frequency is 425 Hz higher than 1700Hz and thus a packet signal on the standard calling frequency of 14,103 KHz would appear at 14,103.43 Khz using the TNC-220. Your transmissions will be readable whenever you are able to copy the other station, regardless of his actual modulation tones. Incidentally, the either upper or lower sideband may be chosen at will due to the nature of the data, although it is conventional to use LSB. Use of USB will merely change the displayed frequency.

The six pole HF filter in the TNC-220 has a 3db bandwidth of 300 Hz. However, additional performance improvements may be obtained by matching the IF bandwidth of your receiver to the data rate you will operate. For 300 baud HF operation, a filter of about 500 Hz bandwidth is optimum; any CW filter wider than this should be better than most SSB filters! The HF tone pair used by the TNC-220 match those used for Radio Teletype (RTTY) allowing the use of built-in RTTY or FSK filters in your transceiver.

Table 5-3. Modem Tone Parameters.

	Bell 202	Bell 103	V.23 Mode 1
Default Data rate	1200 bauds	300 bauds	None
Shift	1000 Hz	200 Hz	400 Hz
Center Freq	1700 Hz	2125 Hz	1500 Hz
Upper Freq	2200 Hz	2225 Hz	1700 Hz
Lower Freq	1200 Hz	2025 Hz	1300 Hz

Multi-Connect Guide

This section is a very brief tutorial on the use of the multiple-connection capability of the TNC-220. Multiple connection capability is a very powerful addition to your TNC-220. It is very useful for traffic net operation, multi-user bulletin boards, path checking and so forth.

What is Multi-Connect?

Multiple connection operation is not the same as multi-way operation. With multiple connect, you may establish several point-to-point "links" with various stations. Multi-way, which is not available, would enable multiple stations to be simultaneously interconnected to each other, with each station seeing all data passed from any station in the group, error free.

Multiple connection operation is another step on the road to proper networking, and networking should eventually allow multi-way operation.

What Commands Set the TNC-220 to Normal Operation?

The TNC-220 defaults the multi-connect-related commands to the following parameters:

CONPERM	OFF
STREAMCAll	OFF
STREAMDouBLe	OFF
STREAMSWitch	|
USERS	1

This sets up the TNC-220 to act just like a TNC does that doesn´t support multiple connections. The key to obtaining this traditional operation is to set **USERS** 1.

How is Multi-Connect Invoked?

If **USERS** is not 1, the TNC-220 will to allow multiple connections to your stations from other stations. In addition, **TRANSPARENT** mode will operate differently, in that incoming data will be prefixed with the current **STREAMSW**itch character and identifier (such as "|A"). Thus, truly transparent operation is not possible with this software release supporting multiple connections.

The **STREAMSW**itch character, defaulted to "|", should be set to a character you won´t normally use. Note that this character may be set to a hex value between $80 and $FF. This may allow you to use 8-bit characters (**AWLEN** 8) if your terminal or computer is capable of generating such "characters." This could help prevent confusion in interpreting incoming data from other stations if they happen to send data that includes your selected **STREAMSW**itch character. When operating a single user MailBox system, it is highly recommended that **STREAMSW** be set to $00.

The **RXBLOCK** command should be invoked when the TNC-220 is serving a host computer. RXBLOCK is designed for automated operations, such as packet bulletin board stations. It is intended to help such systems discriminate between data received from the connected station and TNC-generated messages. Although not foolfproof, enabling **STREAMDBL** may also help in sorting out **STREAMSW**itch characters included in the received data from a valid stream switches generated by your TNC-220.

STREAMCAll should be especially helpful when manually operating a station in which you allow multiple connections. This command prefixes each new stream of data from the TNC with the stream identification.

When in **COMMAND** mode the stream may be switched by entering the **STREAMSW**itch character (default "|"), followed by a stream idenţifier ("A" through "J"), followed by the command you wish to give.

```
cmd:|E C
|E Connect in Progress
cmd:
```

When in **CONVERSE** mode, you may switch streams by entering the **STREAMSWitch** character (default "|"), followed by a stream identifier ("A" through "J"), followed by the data you wish to send to the station on that stream. See the example in the description of **STREAMCAll** for an illustration of this.

Chapter 6

Pac-Comm TNC-220 COMMANDS and MESSAGES

This chapter serves as a complete reference to all TNC-220 commands. Commands are used to control the many variable values which affect your TNC-220's operation, as well as causing it to perform specific actions, such as connecting to another station to start a QSO. You can enter a command to your TNC-220 whenever you see the command-mode prompt:

cmd:

You will change parameters and issue instructions to the TNC-220 by typing commands composed of English-like words or word abbreviations, which are called keywords, and variables which are numbers or strings of characters chosen by the user. You will probably never change some of these parameters; however, one of Pac-Comm's goals is to allow each user maximum flexibility to adapt the TNC-220 to his environment.

Entering Commands

You may use either upper case (capital letters) or lower case (small letters) when you enter commands. In order to have your TNC-220 accept a command line, you must finish the line with a <CR>, or carriage return character. This won't be mentioned explicitly in the examples below. Before you type the final <CR> of your command, you can correct typing mistakes or cancel the line completely. See Chapter 4 for a discussion of input line editing.

This chapter will use UPPER CASE for commands and lower case for explanatory text. In examples showing input typed by the user together with the responses of the TNC-220, the user's input will be shown in **bold face**.

Whenever the TNC-220 accepts a command which changes a value, it will display the previous value. For example, if you type

cmd:**XFLOW OFF**

you might see the display

XFLOW was ON
cmd:

This reminds you of what you have done, and indicates that the value has been successfully changed.

If you type something your TNC-220 can't understand you will get an error message. If you type an unrecognized command, you will see the message ?EH. If you get a command name correct, but the arguments are wrong, you will see the message ?BAD. A complete list of error messages appears at the end of this chapter. For example,

```
cmd:ASDFASDF        !! This isn't a command.
?EH
cmd:BEACON E        !! A parameter was left out.
?BAD
```

Command List

The commands are listed alphabetically, and each command entry contains several sections, as follows.

COMMAND NAME

The boldface command name at the top of the entry is the word you will type in order to have your TNC-220 execute this command. The command name is printed with some letters underlined. These letters form the minimum abbreviation that you may use and still have your TNC-220 understand your command. Of course, you may type out the entire command word, or any abbreviation longer than the minimum abbreviation, if you like.

For example, the command **MYCALL** may be specified by simply typing **MY**. The abbreviation M is not sufficient (and will be interpreted as a different command), but **MY**, **MYC**, **MYCA**, **MYCAL** or **MYCALL** are all acceptable.

If the command requires parameters, they will be shown after the command name.

For commands that set values, your TNC-220 assumes a "most often used" or default condition. The defaults are the values stored in EPROM which are loaded into RAM when the system is first powered up, or when you give the RESET command. Immediate commands perform actions rather than setting values, and don't have defaults.

Parameters

There are several types of parameters. Some parameters can have one of only two values, such as ON and OFF or EVERY and AFTER. If a parameter must be one of two values, the choices are shown separated by a vertical bar. You may use YES instead of ON and NO instead of OFF.

A parameter designated as **n** is a numeric value. These values may be entered as ordinary decimal numbers, or as hexadecimal, or "hex", numbers by preceding the number with a $ symbol. When the TNC-220 shows some of these numeric parameters (those which set special characters), they will be given in hex. The "digits" of a hex number represent powers of 16, analogous to the powers of 10 represent ed by a decimal number. The numbers 10 through 15 are denoted by the hex digits A through F.

For example,

$1B = 1*16 + 11 = 27
$120 = 1*16*16 + 2*16 + 0 = 288

A parameter designated as **text**, such as the argument to **CTEXT**, may be entered in upper or lower case, and may include numbers, spaces, and punctuation. The text is accepted exactly as typed by the user.

Several commands require **callsigns** as parameters. While these parameters are normally Amateur call signs, they may actually be any collection of numbers and at least one letter up to six characters; they are used to identify stations sending and receiving packets. A callsign may additionally include sub-station ID (SSID), which is a decimal number from 0 to 15 used to distinguish two or more stations on the air with the same Amateur call (such as a base station and a repeater). The call sign and SSID are entered and displayed as call-n, e.g., K0PFX-3. If the SSID is not entered, it is set to 0, and SSIDs of 0 are not displayed by the TNC-220.

Some commands have parameters which are actually lists of items. For example, you may specify as many as eight callsigns to be selectively monitored with the command LCALLS. The second and later items in the list are optional, and you may separate the list items with blank spaces or with commas.

These examples may help you to understand the explanations above.

BEACON EVERY|AFTER n

means that the command BEACON requires an argument which must be either **EVERY** or **AFTER** (abbreviated to E or A), and an argument n which the user may choose from a range of values. An acceptable command might be **BEACON** E 2.

CONNECT call#1 [VIA call#2[,call#3...,call#9]]

means that the command **CONNECT** requires a callsign argument **call#1**. You may optionally include the keyword **VIA**, followed by a list of one to eight callsigns, call2 through **call#9**. The **callsigns** in the list, if included, must be separated by commas (as shown), or by blank spaces. An acceptable command might be C **N2WX V AD7I WB9FLW**.

You can see the current value of the command's arguments by typing the command name by itself, without any arguments. For example,

```
cmd:CONOK Y             !! Sets the value to YES (ON).
CONOK was OFF           !! Displays previous value.
cmd:CONOK               !! Command with no arguments
CONOK ON                !! Displays present value.
```

The command, **DISPLAY**, shows the values of all parameters or groups of parameters.

ALPHABETIC LISTING OF COMMANDS, WITH EXPLANATIONS

This section describes the command's action and the meaning of each argument. Examples may be included of situations in which the command might be used. See also the **COMMAND REFERENCE CARD** for a convenient summary of commands and default values.

8BITCONV ON|OFF Default: OFF

Parameters: ON - The high-order bit is not stripped in Converse Mode.
OFF - The high-order bit is stripped in Converse Mode.

This command enables transmission of 8-bit data in Converse Mode. If **8BITCONV** is **OFF**, the high-order bit (bit 7) of characters received from the terminal is removed before the characters are transmitted in a packet. The standard ASCII character set requires only 7 bits, and the final bit is used as a parity bit or ignored. Setting bit 7 in text characters transmitted over the air may cause confusion at the other end.

If you need to transmit 8-bit data, but don't want all the features of Transparent Mode, you should set **8BITCONV ON** and **AWLEN 8**. This may be useful if you are using a special non-ASCII character set.

Bit 7 is always removed in Command Mode, since commands require only the standard 7-bit ASCII character set.

AUTOLF ON|OFF Default: ON

Parameters: ON - A linefeed character (<LF>) is sent to the terminal after each carriage return character (<CR>).
OFF - A <LF> is not sent to the terminal after each <CR>.

AUTOLF controls the display of carriage return characters received in packets as well as echoing those that are typed in.

If the TNC-220's sign-on message lines appear to be typed over each other, you should set **AUTOLF ON**. If the TNC-220's sign-on message appears to be double-spaced, you should set **AUTOLF OFF**. If the TNC-220's sign-on message appears to be single-spaced, you have **AUTOLF** set correctly.

This command only affects what is displayed, not the data sent in packets. If you want to add linefeed characters to outgoing packets, use the command **LFADD**.

AWLEN n Default: 7

Parameters: n = 7 - 8, specifying the number of data bits per word.

This value defines the word length used by the serial IO terminal port.

For most packet operations, including conversation, bulletin board operation, and transmission of ASCII files, you should set **AWLEN 7**. If 8 bit words are transmitted to the TNC-220 in Command Mode or Converse Mode, the eighth bit is normally removed, leaving a standard ASCII character, regardless of the setting of **AWLEN**.

To transmit and receive packets retaining all 8 data bits of each character, as you need to do if you send executable files or other special data, you should use Transparent Mode and set **AWLEN 8**. Alternatively, you can use Converse Mode and set **AWLEN 8** and **8BITCONV ON** (however, the data you then send must handle the Converse Mode special characters with the **PASS** prefix).

AX25L2V2 ON|OFF Default: ON

Parameters: ON - The TNC-220 uses AX.25 Level 2 Version 2.0 protocol.
OFF - The TNC-220 uses AX.25 Level 2 Version 1.0 protocol.

Some implementations of the earlier version of AX.25 protocol (e.g., TAPR's TNC 1) won't properly digipeat version 2.0 AX.25 packets. This command exists to provide compatibility with these other TNCs until their software has been updated. During the protocol transition period, you should set **AX25L2V2 OFF**. After your local area TNCs are updated to the newer protocol version, you should set **AX25L2V2 ON**.

AXDELAY n Default: 0

Parameters: n = 0 - 180, setting voice repeater keyup delay in 10 ms increments.

AXDELAY specifies a period of time the TNC-220 is to wait, in addition to the normal delay set by **TXDELAY**, after keying the transmitter and before data is sent. This feature will be used by groups using a standard "voice" repeater to extend the range of the local area network. Repeaters with slow mechanical relays, split sites, or other circuits which delay transmission after the RF carrier is present require some time to get RF on the air.

If you are using a repeater that hasn't been used for packet operations before, you will have to experiment to find the best value for **n**. If other packet stations have been using the repeater, check with them for the proper setting. Note that this command acts in conjunction with **AXHANG**.

Note that the TAPR TNC 1 and other TNCs using version 3.x firmware interpret **n** in 120 ms intervals. The value set by **AXDELAY** on TNC-220 will thus be 12 times the value used by a TNC 1 user to give the same delay time.

AXHANG n Default: 0

Parameters: n = 0 - 20, setting voice repeater hang time in 100 ms increments.

This value can be used to increase channel efficiency when an audio repeater with a hang time greater than 100 ms is used. For a repeater with a long hang time, it is not necessary to wait for the repeater keyup delay after keying the transmitter if the repeater is still transmitting. If the TNC-220 has heard a packet sent within the hang period, it will not add the repeater keyup delay (**AXDELAY**) to the keyup time.

If you are using a repeater that hasn't been used for packet operations before, you will have to experiment to find the best value for **n**. If other packet stations have been using the repeater, check with them for the proper setting.

Note that the TAPR TNC 1 and other TNCs using version 3.x firmware interpret **n** in 120 ms intervals. The value set on the TNC-220 for **AXHANG** will be 6/5 the value used by a TNC 1 user for the same hang time (when converting, round down to the nearest integer).

BAUD

BAUD is an immediate command which displays the current terminal and radio baud rates and presents a selection menu. This command may be used at any time without changing the link state. The display appears as follows:

```
Radio   : 300
Terminal: 1200

Baud Rates
Radio    Terminal
A-300      G-300
B-600      H-1200
C-1200     I-2400
D-2400     J-4800
E-4800     K-9600
F-9600

Q - Make changes
Select?
```

The initial portion of the **BAUD** display lists the current data transmission rate of both the terminal port and the radio port which is in use. The second portion of the display lists all supported radio and terminal baud rates and gives a command letter for each. The final portion of the display is the word **Select?**. If you wish to change either the radio or terminal baud rate, enter the letter which corresponds to the desired baud rate. The menu will reappear to allow additional selections. When you have made your final baud rate changes, respond to **Select?** with a **Q** to quit the BAUD function.

If JP4 is ON, the radio portion of the **BAUD** command will not function since radio port baud rates are under program control.

Baud rate selections made with the **BAUD** command are preserved in battery backed RAM.

BEACON EVERY|AFTER n   Default: EVERY 0

Parameters: EVERY - Send beacon at regular intervals.
AFTER - Send beacon once after the specified time interval with no packet activity.
n = 0 - 250, specifying beacon timing in 10 second intervals. A value of 0 disables the beacon.

This command enables beacon sending and causes the first beacon frame to be transmitted. A beacon frame consists of the text specified by **BTEXT** in a packet addressed to "BEACON" and sent via the digipeat addresses specified by the **UNPROTO** command, if any. NOTE: If **BTEXT** is blank, no beacons will be sent.

If the keyword **EVERY** is specified, a beacon packet is sent every **n*10** seconds. This mode might be used to transmit packets for testing purposes.

If **AFTER** is specified, a beacon is sent only after **n*10** seconds have passed with no packet activity. In this case, the beacon is sent only once until further activity is detected. This mode can be used to send announcements or test messages only when packet stations are on the air. If you choose **n** properly you can avoid cluttering a busy channel with unnecessary transmissions.

Beacon frames from other TNCs can be monitored by setting **MONITOR ON**.

BKONDEL ON|OFF   Default: ON

Parameters: ON - The sequence <BACKSPACE> <SPACE> <BACKSPACE> is echoed when a character is deleted from the input line.
OFF - The <BACKSLASH> (\) is echoed when a character is deleted.

This command determines the way the display is updated to reflect a character deletion in Command Mode or Converse Mode.

The <BACKSPACE> <SPACE> <BACKSPACE> sequence will properly update the screen of a video display. If you have a video display terminal or computer, you should set **BKONDEL ON**.

The <BACKSPACE> <SPACE> <BACKSPACE> sequence on a printing terminal would result in overtyped text. If you have a paper-output display, or if your terminal does not respond to the <BACKSPACE> character (<CTRL-H>), you should set **BKONDEL OFF**. The TNC-220 will type a <BACKSLASH> for each character you delete. You can display the corrected input line by typing the redisplay-line character, which is set by the command **REDISPLA**.

BTEXT text   Default: blank

Parameters: text - Any combination of characters and spaces, up to a maximum length of 120 characters.

BTEXT specifies the content of the data portion of a beacon packet. The default text is an empty string, i.e., no message. Beacon packets are discussed in more detail under the **BEACON** command.

You can send multiple-line messages in your beacon by including carriage return (<CR>) characters in the text. The <CR> character can be included by preceding it with the pass character. The pass character is set by the **PASS** command. If you enter a text string longer than 120 characters, an error message will appear and the command will be ignored.

For example, a Bulletin Board program might set the beacon **text** to a message like this, updating the **text** after each connection:

Mailbox on line. Messages for WB9FLW, AD7I, K9NG.

To clear the **BTEXT text** without issuing the **RESET** command, use a % or & character as the first character in the text. No beacons will be sent if BTEXT is null.

BUDLIST ON|OFF Default: OFF

Parameters: ON - Ignore frames from stations which are not in the LCALLS list.
OFF - Ignore frame from stations which are in the LCALLS list.

BUDLIST works in conjunction with the command **LCALLS**, which sets up a callsign list. These commands determine which packets will be displayed when you have set **MONITOR ON**. **BUDLIST** specifies whether the callsigns in the list are the ones you want to ignore or, alternatively, are the only ones you want to listen to.

If you want to listen only for packets from a limited list, you should enter this list with **LCALLS** and set **BUDLIST ON**. You can use this feature, for example, to have your TNC-220 "keep an ear out" for a particular station while you converse with someone else.

If you want to ignore (including restricting digipeating and connecting) packets from a limited list, you should list the callsigns to ignore in **LCALLS** and set **BUDLIST OFF**. For example, if there is a bulletin board on frequency, you can ignore it while monitoring other conversations.

CALIBRA

CALIBRA is an immediate command, and is used to transfer control to the modem tone generation routine. The **CALIBRA** command may be performed at any time without altering the current link state. Uses for the calibration tones are discussed in Chapter 3.

Briefly, the commands available in the calibration routine are:

<SPACE> Switch the transmit audio to the other tone.
K Toggle PTT line on/off.
Q Quit calibration routine.

CANLINE n Default: $18 <CTRL-X>

Parameters: n = 0 - $7F, specifying an ASCII character code.

This command is used to change the cancel-line input editing command character. The parameter **n** is the ASCII code for the character you want to type in order to cancel an input line. You can enter the code in either hex or decimal.

For example, to change the cancel-line character to <CTRL-U>, you would set **CANLINE** $15 or **CANLINE** 21.

If you cancel an input line in Command Mode, the line will be terminated with a <BACKSLASH> character, and you will see a new **cmd:** prompt. If you cancel a line in Converse Mode, you will see only the <BACKSLASH> and a new line. You can cancel only the line you are currently typing. Once a <CR> has been entered, you can not cancel an input line using the cancel-line character. Note that if your send-packet character is not <CR>, the cancel-line character will cancel only the last line of a multi-line packet. To cancel the entire packet, use the **CANPAC** character. Line cancellation, like all other input editing features, is disabled in Transparent Mode.

CANPAC n Default: $19 <CTRL-Y>

Parameters: n = 0 - $7F, specifying an ASCII character code.

CANPAC is used to change the cancel-packet input editing command character. The parameter **n** is the ASCII code for the character you want to type in order to cancel an input packet. You can enter the code in either hex or decimal.

If you cancel a packet in Converse Mode, the line will be terminated with a <BACKSLASH> character and a new line. You can only cancel the packet that is currently being entered. Once you have typed the send-packet character, or waited **PACTIME** (if **CPACTIME** enabled), the packet can not be cancelled even if it has not been transmitted.

Packet cancellation, like other input editing features, is disabled in Transparent Mode.

The cancel-packet character also functions to cancel display output in Command Mode. If you are in Command Mode and type the cancel-packet character, any characters that would be typed on the screen (except echoed characters) are "thrown away" by the TNC-220. Typing the cancel-output character a second time restores normal output. To see how this works, try typing **DISPLAY**, then type a <CTRL-Y>. The command list display will stop. You won't see any response from the TNC-220 to commands. Now type another <CTRL-Y>, and type

DISPLAY again to see that the display is back to normal.

You can use the cancel-display feature if you inadvertently do something that causes the TNC-220 to generate large amounts of output to the terminal, such as giving the **DISPLAY** command or setting **TRACE ON**. If you are in Converse Mode or Transparent Mode and want to cancel display output, you must exit to Command Mode and then type the cancel-packet character.

CBELL ON|OFF Default: OFF

Parameters: ON - A bell character will be sent to the terminal as part of the connect message.
OFF - No bell character is included in the connect message.

This command is used to control whether an ASCII $07 (BELL) character is sent as part of the connected message. When set ON, the bell character immediately precedes the asterisk portion of the connected indication.

This command is not related to the **MFILTER** command. **CBELL** will produce a 'CONNECT BELL' even if **MFILTER** is set to delete incoming $07 (BELL) characters.

CHECK n Default: 30

Parameters: n = 0 - 250, specifying the check time in 10 second intervals. A value of 0 disables this feature.

This command sets a connection timeout. When your TNC-220 and another station are connected, and the other station "disappears," your TNC-220 could remain in the connected state indefinitely, refusing connections from other stations. This might happen if propagation changes or an intermediate digipeater station is turned off. In order to prevent this lockup, the TNC-220 will try to clean up the link if the specified time elapses without any packets being heard from the other TNC. This feature depends on the setting of **AX25L2V2**.

If **AX25L2V2** is ON, the TNC-220 will send a "check packet" to verify the presence of the other station if no packets have been heard from it for **n*10** seconds. This frame contains no information, but is interpreted by the receiving station's TNC as an inquiry as to whether it is still connected. If the receiving TNC is still connected, it sends an appropriate response packet. If the TNC-220 initiating the inquiry does not hear a response after RETRY+1 attempts, it commences a disconnect sequence, as if the **DISCONNE** command had been given.

If **AX25L2V2** is **OFF** and the other station has not been heard for **n*10** seconds, the TNC-220 will not attempt an inquiry, but will send a disconnect packet, just as if you had typed the command **DISCONNE**.

CLKADJ n Default: 0

Parameters: n = 0 - 65535, specifying the correction factor to be applied to the real-time clock routine.

A value of "0" is a special case and means no correction factor will be applied. If the value of **CLKADJ** is non-zero, then the correction factor is calculated as:

relative clock speed in % = 100 - (9.16667 * 1/n)

The real-time clock routine is used to keep track of year, month, day, hour, minute and second as specified in the **DAYTIME** command. It should be noted that the real-time clock is not intended to be your ham-shack reference clock, but is useful for approximate time stamping information.

CMDTIME n

Parameters: n = 0 - 250, specifying Transparent Mode timeout value in 1 second intervals. If n is zero, the only exit from Transparent Mode is to send a BREAK signal or interrupt power to the TNC-220.
Default: 1

This command sets the Transparent Mode timeout value. In order to allow escape to Command Mode from Transparent Mode while permitting any character to be sent as data, a guard time of **n** seconds is set up.

The same Command Mode entry character used for exit from Converse Mode is used to exit Transparent Mode, but the procedure is different. (The Command Mode entry character is set by **COMMAND**.) Three Command Mode entry characters must be entered less than **n** seconds apart, with no intervening characters, after a delay of **n** seconds since the last characters were typed. After a final delay of n seconds, the TNC-220 will exit Transparent Mode and enter Command Mode. You should then see the prompt "cmd:"

The diagram below illustrates this timing.

```
   last            first           second          third          TNC-220
   terminal        command         command         command        now in
   input           mode            mode            mode           Command
     |             entry           entry           entry          Mode
     |             character       character       character        |
     |               |               |               |              |
     |<--longer than n-->|<-shorter than n->|<-shorter than n->|<-----n----->|
```

CMSG ON|OFF Default: OFF

Parameters: ON - A text message is sent as the first packet after a connection is established.
OFF - The text message is not sent.

CMSG enables automatic sending of the message set by **CTEXT** whenever your TNC-220 accepts a connect request from another TNC.

For example, if you have left your station running even though you don't want to operate just now, you might want to set **CMSG ON** to let people know that you can't talk when they connect to your TNC-220. When you are ready to operate, you would set **CMSG OFF**.

CMSGDISC ON|OFF Default: OFF

Parameters: ON - The TNC-220 will initiate a disconnect immediately after acknowlegement of the **CTEXT** sent by the **CMSG** feature.
OFF - The TNC-220 will not initiate a disconnect connected with **CMSG**

This command controls whether TNC-220 will initiate a disconnect after the packet containing connect text (CTEXT) is acknowledged.

If **CMSG** is OFF, or **CTEXT** has no connected text, the TNC-220 initiates its disconnect immediately upon receiving information or acknowledgement frames from the other station. This command may be useful to bulletin board operators or others with a need to send a short message, confirm it's receipt, and disconnect.

Use this command with care - If you find you're able to receive connects, yet never get data, it's possible **CMSGDISC** has been left on.

COMMAND n Default: $03 <CTRL-C>

Parameters: n = 0 - $7F, specifying an ASCII character code.

This command is used to change the Command Mode entry character. You can enter the code in either hex or decimal.

Command Mode is entered from Converse Mode when this character is typed. If you type the Command Mode entry character while you are already in Command Mode, nothing will happen. To see how this works, enter Converse Mode by typing **CONVERS**. Anything you type will become packet data. Now type a <CTRL-C>. You will see the Command Mode prompt, indicating successful exit to Command Mode. The display might look like this:

```
cmd:CONVERS
Hello World! I'm on the air on packet radio!
[enter <CTRL-C>]
cmd:
```

See the CMDTIME entry or the Chapter 5 discussion of Transparent Mode for information on how the Command Mode entry character is used for escape from Transparent Mode.

CONMODE CONVERS|TRANS Default: CONVERS

Parameters: CONVERS - Sets automatic entry to Converse Mode when a connection is established.
TRANS - Sets automatic entry to Transparent Mode when a connection is established.

CONMODE controls which mode the TNC-220 will be placed in after a connection. The connection may result either from a connect request received over the air or a keyboard command. For most operations, you would set **CONMODE** to **CONVERS**. However, if you are using Transparent Mode for a bulletin board program, for example, you would set **CONMODE** to **TRANS** so that the correct mode will be entered when your bulletin board receives a connect request.

If you initiate a connection with the **CONNECT** command, the timing of the entry into Converse or Transparent Mode is determined by **NEWMODE**.

If the TNC-220 is already in Converse or Transparent Mode when the connection is completed, the mode will not be changed. If you have typed part of a command line when the connection is completed, the mode change will not take place until you complete the command or cancel the line. This prevents the last part of your command from inadvertently being sent as a packet.

CONNECT call#1 [VIA call#2[,call#3...,call#9]]

Parameters: call #1 - **Callsign** of TNC to be connected to.
call #2 - Optional **callsign** of TNC to be digipeated through. As many as eight digipeat addresses can be specified.

The part of the command line in brackets, **VIA call#2[, call#3...,call#9]** is optional. The double-bracketed text, **,call#3...,call#9**, would only be used if **VIA call#2** is present. The brackets are not typed.

Each **callsign** may include an optional sub-station ID specified as -n immediately following the **callsign**. The digipeat fields are specified in the order in which you want them to relay the packets to the destination, **call1**.

CONNECT is an immediate command. It initiates a connect request to TNC **call1**, optionally through digipeaters. If **NEWMODE** is **ON**, the TNC-220 will immediately enter Converse Mode or Transparent Mode, as specified by the command **CONMODE**. If **NEWMODE** is **OFF**, the TNC-220 will enter Converse Mode or Transparent Mode when the connection is successfully completed.

An error message is returned if the TNC-220 is in a connected state, or is already attempting to connect or disconnect. If no response to the connect request occurs after the number of attempts specified by **RETRY**, the command is aborted and a message is typed. The TNC-220 returns to Command Mode if **NEWMODE** is **ON**. If **NEWMODE** is **OFF**, the mode does not change, i.e. the TNC-220 remains in Command Mode. For example, to connect to WA7GXD using N0ADI-1 (nearby) and WD0ETZ (nearby WA7GXD) as digipeaters, type:

```
CONNECT WA7GXD VIA N0ADI-1, WD0ETZ
```

Packets coming back from WA7GXD access the digipeaters in the opposite order. Thus, packets from WA7GXD will first be repeated by WD0ETZ, then by N0ADI-1.

CONOK ON|OFF Default: ON

Parameters: ON - Connect requests from other TNCs will be accepted.
OFF - Connect requests from other TNCs will be rejected.

This command determines the action taken by the TNC-220 when a connect request for it is received though the radio. If **CONOK** is ON, the request will be acknowledged, the standard connect message will be typed and either Converse or Transparent Mode will be entered, depending on the setting of **CONMODE**.

If **CONOK** is **OFF** and the TNC-220 is not in Transparent Mode, a connect request will display this message.

connect request: <call> (callsign of station attempting to connect)

The TNC-220 will issue a DM packet, or "busy signal" to the requesting station. If your TNC-220 receives a DM packet in response to a connect request, it will display

*** <call> station busy (callsign of station to which connect was attempted)

For example, if you want to leave your station running as a digipeater you might set **CONOK OFF** until you are ready for a conversation. If you get a connect request in the meantime, you can change your mind. Stations attempting to connect to your TNC-220 will be able to see that the station is up but not available for connection (it could still operate as a digipeater).

CONPERM ON|OFF Default: OFF

Parameters: ON - The current connection on the current stream will not be allowed to enter the disconnected state.
OFF - The current stream may be connected to and disconnected from other stations.

This command, when switched ON, forces the TNC-220 to always maintain the current connection, even when frames to the other station exceed **RETRY** attempts to get an acknowledgment. **RESTART** and power off/on cycling will not affect this connected state.

This command only takes effect when a connection is established. It functions on a stream-by-stream basis when multiple connections are allowed.

It is useful for certain networking applications, meteor scatter and other noisy, less-reliable links, while still allowing connections on other streams to operate normally (automatic disconnect based on **RETRY**, etc.).

CONSTAMP ON|OFF Default: OFF

Parameters: ON - Connect status messages are time stamped.
OFF - Connect status messages are not time stamped.

This command enables time stamping of ***** CONNECTED** status messages. The date and time information is then available for use by computer application programs. The date and time must be set initially by the **DAYTIME** command before time stamping will occur. The date format is determined by the **DAYUSA** command.

For example, if **CONSTAMP** is ON, **DAYUSA** is ON, and the date and time have been set, a connect message might appear as follows.

*** CONNECTED to N2WX [05/28/85 16:28:31]

CSTATUS

CSTATUS is an immediate command which shows the stream identifier and link state of all ten streams (links), the current input and output streams, and whether or not each stream is "permanent" (see **CONPERM**).

An example of a display resulting from issuing a CSTATUS command is:

```
cmd:CS
A stream - IO Link state is: CONNECTED to 305MLB
B stream -    Link state is: CONNECTED to AD7I P
C stream -    Link state is: DISCONNECTED
D stream -    Link state is: CONNECTED to N0ADI via K9NG-2
```

The example above shows the A stream is the currently selected or active stream for both input and ouptut, which is indicated by the I and O symbols. The B stream is connected to AD7I "permanently." All other streams' states are shown as they might normally appear with multiple connections.

CONVERS

CONVERS is an immediate command, and will cause the TNC-220 to exit from Command Mode into Converse Mode. Any link connections are not affected. Once in Converse Mode, everything you type is packetized and transmitted over the radio. Typing the Command Mode entry character returns the TNC-220 to Command Mode. See the discussions of Converse Mode in Chapter 4 and Chapter 5. The letter K is an alias for CONVERS allowing convenient single key transfer from the command to converse mode.

CPACTIME ON|OFF Default: OFF

Parameters: ON - Packet timeout is used in Converse Mode.
OFF - Packet timeout is not used in Converse Mode.

This command enables the periodic automatic sending of packets in Converse Mode. After CPACTIME has elapsed since the last frame was transmitted, any characters which have been input from the terminal will be transmitted. The SENDPAC character appearing in the text will also trigger transmission of a packet. This feature may be used for computer communications, such as Bulletin Board operation, when the full Transparent Mode features are not desired.

If CPACTIME is ON, characters are packetized and transmitted periodically as they are in Transparent Mode, but local editing and display features of Converse Mode are enabled, and software control may be used. For a discussion of how periodic packetizing works, see the command PACTIME, which controls the rate and mode of packet assembly.

You should set CR OFF in this mode, since otherwise the send-packet character will be inserted in the data being packetized even though it was not typed. In order to include <CR> characters in transmitted packets, set SENDPAC to a normally unused character (e.g., <CTRL-P>), at which point the TNC-220 will treat <CR> as an ordinary character.

You can set CPACTIME ON for a mode of operation similar to full break-in CW, in which your text is transmitted soon after you type it, but in short bursts of a few characters, and the other station may break in at will. Some operators find it easier to carry on a conversation in this mode, since it eliminates the delays while long packets are being typed.

CR ON|OFF Default: ON

Parameters: ON - The send-packet character, normally <CR>, is appended to all packets sent in Converse Mode.
OFF - The send-packet character is not appended to packets.

When CR is ON, all packets sent in Converse Mode will include, as the last character of the packet, the send-packet character which forces the packet to be sent. If CR is OFF, the send-packet character is interpreted solely as a command to the TNC-220, not as data to be included in the packet, and furthermore, it will not be echoed to the terminal.

Setting CR ON and SENDPAC $0D results in a natural conversation mode. Each line is sent when a <CR> is entered, and arrives at its destination with a <CR> at the end of the line. If the station at the other end reports overprinting of lines on his display, you can set LFADD ON, or the other station can set AUTOLF ON.

CTEXT text Default: blank

Parameters: text - Any combination of characters and spaces, up to a maximum length of 120 characters.

CTEXT specifies the text of the packet to be sent after a connection is made, if **CMSG** is ON. The default **text** is an empty string, i.e., no message.

You can send multiple-line messages by including carriage return (<CR>) characters in the text. The <CR> character can be included by using the pass character (default value CTL-V) immediately preceding it (see the **PASS** command). If you enter a **text** string longer than 120 characters, an error message will appear and the command will be ignored.

To clear the **CTEXT text** without issuing a **RESET** command, use a % or & as the first character in the message.

DAYTIME date & time No default

Parameters: date & time; Current date and time to set.

This command allows you to set the current date and time for the TNC-220. The format for entering the **date & time** is

yymmddhhmm

where **yy** is the last two digits of the year, **mm** is the two-digit month code (01-12), **dd** is date (01-31), **hh** is the hour (00-23), and **mm** is the minutes after the hour (00-59). All these codes must be exactly two digits, so that numbers from 0 to 9 must be entered with leading zeros. The TNC-220 does not check thorough ly for the correct number of days in a month, so you should use some judgment when you set the date.

The **date & time** parameter is used by the commands **CONSTAMP** and **MSTAMP** to "time stamp" received and monitored messages. Entries in the "heard" (displayed by **MHEARD**) are also time stamped if **date&time** has been set. The TNC-220's time is updated continuously as long as it is powered up. You must reset the date and time each time the TNC-220 is powered on or else the commands **CONSTAMP** and **MSTAMP** will not enable time stamping.

If you type **DAYTIME** without a parameter, the TNC-220 will display the current date and time. The format of the display is **dd-mm-yy hh:mm** if **DAYUSA** is **OFF**, and **mm/dd/yy hh:mm** if **DAYUSA** is ON. The format for entering **date&time** is not affected. For example,

```
cmd:DAYTIME 8606161530
```

sets the date and time to June 16, 1986 at 3:30 PM. The display of the date and time, (**DAYUSA ON**) would be:

```
cmd:DAYTIME
02/29/84 15:30:26
```

DAYUSA ON|OFF Default: ON

Parameters: ON - Date is displayed in the format **mm/dd/yy**.
OFF - Date is displayed in the format **dd-mm-yy**.

This command determines the format for the TNC-220's display of the date. If **DAYUSA** is **ON**, the standard U.S. format is used; if **DAYUSA** is **OFF**, the standard European format is used. This command affects the format of the date display used in "time stamps" as well as the display when **DAYTIME** is entered without parameters. The format for entering the time using **DAYTIME** is not affected.

For example, if **DAYUSA** is **ON**, then July 2, 1984 at 9:28:44 AM would be displayed as

```
cmd:DAYTIME
07/02/84 9:28:44
```

If **DAYUSA** is **OFF** the same date and time would appear as

```
cmd:DAYTIME
02-07-84 9:28:44
```

DELETE ON|OFF Default: OFF

Parameters: ON - The delete character input editing key is <DELETE> ($7F).
OFF - The delete character input editing key is <BACKSPACE> ($08).

This command is used to change the input editing command for character deletion. When this character is typed, the last character from the input line is deleted. How the TNC-220 indicates the deletion is controlled by the **BKONDEL** command.

You can not use this character to delete past the beginning of a line, although you can delete <CR> characters that have been entered in the text with the pass character.

To see a corrected display of the current line after you have deleted characters, type the redisplay-line character, which is set by the **REDISPLA** command.

DIGIPEAT ON|OFF Default: ON

Parameters: ON - The TNC-220 will digipeat packets if requested.
OFF - The TNC-220 will not digipeat packets.

When this parameter is **ON**, any packet received that has your TNC-220's callsign (including SSID) in the digipeat list of its address field will be retransmitted. Each station included in the digipeat list relays the packet in its turn, marking the packet so that it will not accidentally relay it twice (unless so requested), and so that the stations will relay the packet in the correct order. Digipeating takes place concurrently with other TNC-220 operations and does not interfere with normal packet operation.

In the spirit of cooperation typical of Amateur operation, you will probably want to set **DIGIPEAT ON** most of the time. However, you might want to disable digipeating if you're not home, or if your transmit relay makes enough noise to wake you up at night.

The command **HID** enables automatic transmission of identification packets if your station is acting as a digipeater.

DISCONNECT

DISCONNECT is an immediate command. It will initiate a disconnect request with the currently connected station. A successful disconnect results in the display:

```
*** DISCONNECTED
```

Other commands may be entered while the disconnect is taking place, although connects are disallowed until the disconnect is completed. If the retry count is exceeded while waiting for the other side to acknowledge, the TNC-220 moves to the disconnected state. If a disconnect command is entered while the TNC-220 is disconnecting, the retry count is immediately set to the maximum number. In either case, the disconnect message is

```
*** retry count exceeded
*** DISCONNECTED
```

Disconnect messages are not displayed when the TNC-220 is in Transparent Mode.

DISPLAY [class]

Parameters: class - Optional parameter-class identifier, one of the following:

ASYNC	display asynchronous port parameters
CHARACTE	display special characters
HEALTH	display health counters and LED status
ID	display ID parameters
LINK	display link parameters
MONITOR	display monitor parameters
TIMING	display timing parameters

DISPLAY is an immediate command, and with no **class** parameter will cause all control parameters and their current values to be displayed. Sub-groups of related parameters can be displayed by specifying the optional parameter-class. Individual parameters can be displayed by entering the parameter name with no options.

DWAIT n Default: 16

Parameters: n = 0 - 250, specifying default wait time in 10 ms intervals.

This value is used to avoid collisions with digipeated packets. The TNC-220 will wait the default wait time after last hearing data on the channel before it begins its keyup sequence, unless it is waiting to transmit digipeated packets. This value should be agreed on by all members of a local area when digipeaters are used in the area. The best value will be determined by experimentation, but will be a function of the keyup time (TXDELAY) of the digipeater stations.

This feature is intended to help alleviate the drastic reduction of throughput that occurs on a channel when digipeated packets suffer collisions. It is necessary because digipeated packets are not retried by the digipeater, but must be restarted by the originating station. If all stations specify a default wait time, and the right value of n is chosen, the digipeater will capture the frequency every time it has data to send, since digipeated packets are sent without this delay.

ECHO ON|OFF Default: ON

Parameters: ON - Characters received from the computer or terminal are echoed by the TNC-220.
OFF - Characters are not echoed.

This command controls local echoing by the TNC-220 when it is in Command or Converse Mode. Local echoing is disabled in Transparent Mode.

If you don't see what you type, you should set **ECHO ON**. If you see two copies of every character you type, you should set **ECHO OFF**. If you see the characters you type displayed correctly, you have ECHO set correctly.

ESCAPE ON|OFF Default: OFF

Parameters: ON - The <ESCAPE> character ($1B) is output as "$" ($24).
OFF - The <ESCAPE> character is output as <ESCAPE> ($1B).

This command specifies the character which will be output when an <ESCAPE> character is to be sent to the terminal. The <ESCAPE> translation is disabled in Transparent Mode.

This command is provided because some terminals, and computer programs that emulate such terminals, interpret the <ESCAPE> character as a special command prefix. Such terminals may alter their displays depending on the characters following the <ESCAPE>. If you have such a terminal, you can protect yourself from unexpected text sequences and from other packeteers by setting **ESCAPE ON**.

See also the **MFILTER** command, which allows general character stripping (rather than character translation) in monitored packets.

FLOW ON|OFF Default: ON

Parameters: ON - Type-in flow control is active.
OFF - Type-in flow control is disabled.

When type-in flow control is enabled, any character entered from the terminal will halt output to the terminal until: (1) a packet is forced (in Converse Mode); (2) a line is completed (in Command Mode); (3) the packet length is exceeded; or (3) the terminal output buffer fills up. Cancelling the current command or packet or typing the redisplay-line character will also cause output to resume. Type-in flow control is not used in Transparent Mode.

Setting **FLOW** ON will keep received data from interfering with data entry. If you (and the person you are talking to) wait for a packet from the other end before starting to respond, you can set **FLOW OFF**. Some Bulletin Board programs may work best with **FLOW OFF**. Some computers with "software UARTs" may be unable to send and receive data at the same time; users of such computers should set **FLOW ON**.

FRACK n Default: 3

Parameters: n = 1-15, marking frame acknowledgment timeout in 1 second intervals.

After transmitting a packet requiring acknowledgment, the TNC-220 will wait for the frame acknowledgment timeout before incrementing the retry counter and sending the frame again. If the packet address includes relay requests, the Retry interval is adjusted to n * (2*m + 1) where m is the number of intermediate relay stations.

When a retried packet is sent, a random wait time is added to the wait times in use to avoid lockups in which two TNCs repeatedly send packets which collide with each other.

FULLDUP ON|OFF Default: OFF

Parameters: ON - Full duplex mode is enabled.
OFF - Full duplex mode is disabled.

When full duplex mode is disabled, the TNC-220 makes use of the Data Carrier Detect signal from the modem to avoid collisions, and acknowledges multiple packets in a single transmission with a single acknowledgment. When full duplex mode is enabled, the TNC-220 ignores the DCD signal and acknowledges packets individually. The latter mode is useful for full-duplex radio operation, such as through an amateur satellite. It should not be used unless both your station and the station you are communicating with are full-duplex stations.

You may find full-duplex mode useful for some testing operations, such as analog or digital loopback tests.

HEADERLN ON|OFF Default: OFF

Parameters: ON - The header for a monitored packet is printed on a separate line from the packet text.
OFF - The header and packet text of monitored packets are printed on the same line.

This command affects the display format for monitored packets. If **HEADERLN** is **OFF**, the address information is displayed with the packet:

KV7D>N2WX: Go ahead and transfer the file.

If **HEADERLN** is **ON**, the address information is displayed, followed by the packet text on a separate line:

N2WX>KV7D:
Sorry, I´m not quite ready yet.

The second format is especially useful when **MCOM** is set **ON** showing all of the digipeating stations between the source and destination. If you have set **MRPT ON** or enabled **MSTAMP**, you may wish to set **HEADERLN ON**, as the packet header quickly becomes long enough to fill a screen when these functions are active.

HEALLED ON|OFF Default: OFF

Parameters: ON - The TNC will "dither" the CON and STA LEDs.
OFF - The TNC will control the CON and STA LEDs in normal fashion.

This command allows the user to redefine the functions of the two CPU controllable LEDs (i.e. the STAtus and CONnect LEDs).

When **HEALLED** is set **ON**, the two LEDs flash in a seemingly random fashion. At a glance, the user may make a judgement on whether the software has crashed or the processor is halted, since the LEDs will probably not flash if the software is not executing properly.

With **HEALLED** set **OFF**, the LEDs function in their normal fashion, indicating the presence of a connection and whether any unacknowledged frames exist.

HID ON|OFF Default: OFF

Parameters: ON - Enables HDLC identification by a digipeater.
OFF - Disables HDLC identification.

This command is used to enable or disable the sending of identification packets by the TNC-220. If **HID** is **OFF**, the TNC-220 will never send an identification packet. If HID is **ON**, the TNC-220 will send an identification packet every 9.5 minutes if the station is digipeating packets. The **ID** command allows the operator to send a final identification packet if the station is being taken off the air.

An identification consists of an unsequenced I frame whose data field is your station identification. The identification packet is addressed to the "CQ" address set by the **UNPROTO** command. Your station identification is your callsign as set by **MYCALL**, with "/R" appended.

ID

ID is an immediate command. It will send a special identification packet. **ID** can be used to force a final identification packet to be sent as a digipeater station is being taken off the air. The identification packet will be sent only if the digipeater has transmitted since the last automatic identification.

An identification consists of an unsequenced I frame whose data field is your station identification. The identification packet is addressed to the "CQ" address set by the **UNPROTO** command. Your station identification is your callsign as set by **MYCALL**, with "/R" appended.

LCALLS **call#1[,call#2...,call#8]** Default: blank

Parameters: call - **Callsign** list. Up to 8 calls, separated by commas.

Each **callsign** may include an optional sub-station ID specified as **-n** immediately following the call. This command works in conjunction with **BUDLIST** and allows selective control of the response to other packet stations. These two commands determine which packets will be displayed when you have set **MONITOR ON**, which stations will be allowed to connect, and which will be allowed to digipeat through your station. **BUDLIST** specifies whether the **callsigns** in the list are the ones you want to ignore or, alternatively, are the only ones you want to listen to.

If you want to monitor selected callsigns, enter your selected list with **LCALLS** and set **BUDLIST ON**.

If you want to ignore selected packets, you should list the **callsigns** to ignore in **LCALLS** and set **BUDLIST OFF**.

LCOK ON|OFF Default: ON

Parameters: ON - The TNC-220 will send lower case characters to the computer or terminal.
OFF - The TNC-220 will translate lower case characters to upper case.

If **LCOK** is **OFF**, lower case characters will be translated to upper case before being output to the terminal. This case translation is disabled in Transparent Mode. Input characters and echoes are not case translated.

If your computer or terminal does not accept lower case characters it may react badly if the TNC-220 sends such characters to it. This command allows you to translate all lower case characters received in packets, as well as messages from the TNC-220, to upper case.

Since echoes of the characters you type are not translated to upper case, you can use this command to make your display easier to read when you are conversing in connected mode. If both operators set **LCOK OFF**, you can each type your own messages in lower case and see incoming packets displayed in upper case. It will then be easy to distinguish incoming and outgoing lines.

LCSTREAM ON|OFF Default: ON

Parameters: ON - The TNC will convert the character following the STREAMSWITCH character to upper case before processing it.
OFF - The TNC will process the character immediately following the STREAMSWITCH character as it is entered.

When operating multi-connect, the user must enter a stream identifier (default A through J) after the STREAMSWITCH character (default |) to select a new logical stream to send data. Normally, the stream identifier must be in upper case, or an error message will result.

When LCSTREAM is ON, the character immediately following the streamswitch character is converted to upper case before being acted upon. Thus, the case (upper or lower) becomes insignificant.

LFADD ON|OFF Default: OFF

Parameters: ON - A <LF> character is added to outgoing packets following each <CR> transmitted in the packet.
OFF - No <LF> is added to outgoing packets.

This function is similar to **AUTOLF**, except that the <LF> characters are added to outgoing packets rather than to text displayed locally. This feature is included in order to maintain compatibility with other packet radio controllers. If the person you are talking to reports overprinting of packets from your station you should set **LFADD ON**. This character insertion is disabled in Transparent Mode.

LIFGNORE ON|OFF Default: OFF

Parameters: ON - Line Feed characters are ignored in **COMMAND** and **CONVERSE** modes
OFF - Line Feed characters are acted upon in **COMMAND** and **CONVERSE** modes.

This command controls whether TNC-220 responds to ASCII Line Feed ($0A) characters or ignores them in **COMMAND** and **CONVERSE** modes.

When turned on, line feeds are totally ignored except in **TRANSPARENT** mode. When turned off, line feed characters are processed normally.

The **LFIGNORE** command is useful when incoming data has imbedded Line Feed characters that cause double spacing of the data on your screen or printer. NOTE: See **LFADD** command description. The transmitting station may have the **LFADD** command set ON inadvertently.

MALL ON|OFF Default: ON

Parameters: ON - Monitored packets include both "connected" packets and "unconnected" packets.
OFF - Monitored packets include only "unconnected" packets.

This command determines the class of packets which are monitored. If **MALL** is **OFF**, only otherwise eligible packets (as determined by the **BUDLIST** and **LCALLS** commands) sent by other TNCs in the unconnected mode are displayed. This is the normal manner of operation when this TNC-220 is being used to talk to a group of TNCs all of which are unconnected.

If **MALL** is **ON**, all otherwise eligible frames are displayed, including those sent between two other connected TNCs. This mode may be enabled for diagnostic purposes or for "reading the mail."

MAXFRAME n Default: 2

Parameters: n = 1 - 7, signifying a number of packets.

MAXFRAME sets an upper limit on the number of unacknowledged packets which the TNC-220 can have outstanding at any one time. This is also the maximum number of contiguous packets which can be sent during any given transmission. If some but not all of the outstanding packets are acknowledged, a smaller number may be transmitted the next time, or new frames may be included in the retransmission, so that the total unacknowledged does not exceed n.

If you perform file transfers, you should experiment with **MAXFRAME** and **PACLEN**. If the link is good, there is an optimum relationship between the parameters set by these commands so that the maximum number of characters outstanding does not exceed the packet receive buffer space of the TNC receiving the data.

MCOM ON|OFF Default: ON

Parameters: ON - All framesare monitored.
OFF - Only information frames are monitored.

This command enables monitoring of connect, disconnect, and all other supervisory frames when **MONITOR** is ON.

When **MCOM** is **OFF**, only I frames (packets containing user information) will be displayed. When **MCOM** is **ON** all control fields are decoded and invalid ones are marked with ????. For I and S frames, sequence number information is also presented. Frames compatible with the AX.25 Level 2.0 standard are also decoded as to the state of the C and PF bits. For Example:

```
WA7GXD>KV7D <I C S0 R0>:
WA7GXD>KV7D <I C P S1 R0>:
KV7D>WA7GXD <RR R F R2>
KV7D>WA7GXD <I C P S1 R2>:
```

As with other monitor commands, the stations monitored are determined by **BUDLIST** and **LCALLS**.

See Chapter 9, Packet Radio Protocol, for more information about the various supervisory frames.

MCON ON|OFF Default: ON

Parameters: ON - Monitor mode remains active when TNC-220 is connected.
OFF - Monitor mode is off while the TNC-220 is connected.

If **MCON** is ON, the **MONITOR** command will enable monitoring while your TNC-220 is connected to another TNC. If **MCON** is **OFF**, the display of monitored packets is suspended when a connect occurs, and is resumed when the TNC-220 is disconnected.

If you want to see all packets displayed when you are not connected but have such display suppressed when you connect to another station, you should set **MCON OFF**.

MFILTER n1[,n2[,n3[,n4]]] Default: none

Parameters: n = 0 - $7F, specifying an ASCII character code. Up to 4 characters may be specified.

This command allows you to specify characters to be "filtered," or eliminated from monitored packets. The parameters **n1**, n2, etc., are the ASCII codes for the characters you want to filter. You can enter the code in either hex or decimal.

For example, if a <CTRL-L> character causes your screen to be cleared, and you don't want this to happen, you can set **MFILTER** 12. If you also want to eliminate <CTRL-Z> characters, which some computers interpret as end-of-file markers, you can set MFILTER 12,26.

MHCLEAR

MHCLEAR is an immediate command. It causes the list of stations heard to be cleared. Use this command in conjunction with **MHEARD** to keep track of the stations on the air over a given period of time, such as an evening or a week. Clear the list of stations heard when you first begin to monitor the packet activity.

MHEARD

MHEARD is an immediate command. It causes the TNC-220 to display the list of stations that have been heard since the last time the command **MHCLEAR** was given. Stations that are heard via a digipeater are marked with a * in the heard log. If you clear the list of stations heard at the beginning of a session, you can use this command to easily keep track of the stations that are active during that period. The maximum number of heard stations that can be logged is 18. If more stations are heard, earlier entries are discarded. Logging of stations heard is disabled when **PASSALL** is **ON**.

If the **DAYTIME** command has been used to set the date and time, entries in the heard log will be time stamped. For example,

```
cmd:MHEARD
W1BEL-2*  06/09/87  21:06:18
cmd:
```

MONITOR ON|OFF Default: ON

Parameters: ON - Monitoring of packet activity is enabled.
OFF - Monitoring of packet activity is disabled.

If **MONITOR** is **ON** and the TNC-220 is not in Transparent Mode, packets not addressed to your TNC-220 may be displayed. The addresses in the packet are displayed along with the data portion of the packet. For example, N2WX>W5FD-3: I'm ready to transfer the file now. The calls are separated by a ">" and the sub-station ID field (SSID) is displayed if it is other than 0. The **MALL**, **BUDLIST**, and **LCALLS** commands determine which packets are to be monitored. The **MCON** command controls the action of monitor mode when the TNC-220 is connected. All monitor functions are disabled in Transparent Mode.

The format of the monitor display is controlled by **HEADERLN**. If you want to see the station addresses on a separate line from the text, you can set **HEADERLN ON**. **MRPT** enables monitoring of the digipeater route as well as source and destination addresses for each packet. **MSTAMP** includes a time stamp with the addresses if **DAYTIME** has been set.

MRPT ON|OFF Default: ON

Parameters: ON - Display digipeating stations for monitored packets.
OFF - Display only the source and destination stations for monitored packets.

This command affects the way monitored packets are displayed. If **MRPT** is **OFF**, only the originating station and the destination are displayed for monitored packets. If **MRPT** is **ON**, the entire digipeat list is displayed for monitored packets, and stations that have already relayed the packet are indicated with an asterisk.

For example, WB9FLW>AD7I,K9NG*,N2WX-7:Hi Paul.

This packet, sent from WB9FLW to AD7I, has been relayed by K9NG but not by N2WX-7. With **MRPT OFF**, the same packet would be displayed as

WB9FLW>AD7I:Hi Paul.

Setting **MRPT ON** increases the length of the address display, and you may wish to set **HEADERLN ON** as well to display this information on a separate line.

MSTAMP ON|OFF Default: OFF

Parameters: ON - Monitored frames are time stamped.
OFF - Monitored frames are not time stamped.

This command enables time stamping of monitored packets. The date and time information is then available for use for automatic logging of packet activity or other computer applications. The date and time are set initially by the **DAYTIME** command, and the date format is determined by the **DAYUSA** command.

Setting **MSTAMP ON** increases the length of the address display, and you may wish to set **HEADERLN ON** as well to display this information on a separate line.

MYCALL call[-n] Default: NOCALL-0

Parameters: call - Callsign of your TNC-220.
n = 0 - 15, an optionally specified sub-station ID (SSID).

This command tells the TNC-220 what its callsign is. This callsign will be placed in the FROM address field for all packets originated by your TNC-220. It will accept frames with this callsign in the TO field and relay frames with this callsign in the digipeat field. **MYCALL** will also be used for identification packets (see **HID** and **ID**).

The default callsign must be changed for proper operation of the protocols. There should never be more than one station with the same callsign (including SSID) on the air at once. The SSID can be used to distinguish two stations with the same Amateur call. The SSID will be 0 unless explicitly set to another value.

MYALIAS call[-n] Default: <blank>

Parameters: call - Alternate identity of your TNC-220.
n = 0 - 15, an optionally specified sub-station ID (SSID).

This command specifies an alternate callsign (in addition to the callsign specified in MYCALL) for use as a digipeater only.

In some areas, wide coverage digipeaters operators have changed the callsign of their machine to a shorter and (usually) easier to remember identifier. International Civil Aviation Organization (ICAO) location identifiers, sometimes combined with telephone area codes, have been used.

Use of this command permits **HID** to identify normally with the **MYCALL**-specified callsign yet permit an alternate (alias) repeat-only "callsign."

NEWMODE ON|OFF Default: OFF

Parameters: ON - Switching to data transfer mode occurs at the time of the **CONNECT** command and return to command mode is automatic at the time of disconnection.
OFF - Switching to data transfer mode occurs at time of connection and no return to command mode occurs at disconnection.

The NEWMODE command may be used to select the way the TNC-220 behaves when connections are made and broken.

If **NEWMODE** is OFF, the TNC-220 will remain in Command Mode after you issue a **CONNECT** command until a connection is actually established. When the connection is established, the TNC-220 will enter Converse Mode or Transparent Mode, depending on the setting of CONMODE. When the connection is terminated, the TNC-220 remains in Converse or Transparent Mode unless you have forced it to return to Command Mode. This is the same as the behavior of TNC 1 (running version 3.x software) under these conditions.

If **NEWMODE** is ON, the TNC-220 will enter the specified conmode as soon as you issue a **CONNECT** command, without waiting for the connection to be established. Anything typed will be packetized to be transmitted once the connection is complete. When the connection is broken, or if the connect attempt fails, the TNC-220 returns to Command Mode.

If you have a Bulletin Board program designed to work with TNC 1 you should set **NEWMODE OFF** if the program relies on the sequence of actions used by TNC 1. Otherwise, you should choose the settings for NEWMODE and NOMODE that seems most convenient to you.

NOMODE ON|OFF Default: OFF

Parameters: ON - The TNC will only switch modes (command, converse or transparent) upon explicit command.
OFF - The TNC will switch modes in accordance with the setting of NEWMODE.

When **NOMODE** is ON, the TNC will never change between CONVERSE or TRANSPARENT mode to COMMAND mode (or vice-versa) on its own. Only user commands (CONV, TRANS, or ^C) may change the typein mode.

If **NOMODE** is OFF, then automatic mode switching is handled according to the setting of the **NEWMODE** command.

NUCR ON|OFF Default: OFF

Parameters: ON - <NULL> characters are sent to the terminal following <CR> characters.
OFF - <NULL> characters are not sent to the terminal following <CR> characters.

This command causes transmission of <NULL> characters (ASCII code $00), producing an effective delay following any <CR> sent to the terminal. The number of <NULL> characters is determined by the command **NULLS**. This delay is required by some hardcopy terminals. You need to set **NUCR ON** if your terminal misses one or more characters after responding to a <CR>. If this is the case, you will sometimes see overtyped lines.

NULF ON|OFF Default: OFF

Parameters: ON - <NULL> characters are sent to the terminal following <LF> characters.
OFF - <NULL> characters are not sent to the terminal after <LF> characters.

This command causes transmission of <NULL> characters (ASCII code $00), producing an effective delay following any <LF> sent to the terminal. The number of <NULL> characters is determined by the command **NULLS**. This delay is required by some display terminals. You need to set **NULF ON** if your terminal sometimes misses characters at the beginning of the line.

NULLS n Default: 0

Parameters: n = 0 - 30, specifying the number of <NULL> characters to send after <CR> or <LF>.

This command specifies the number of <NULL> characters (ASCII code $00) to send to the terminal after a <CR> or <LF> is sent. In addition to setting this parameter value, **NUCR** and/or **NULF** must be set to indicate whether nulls are to be sent after <CR>, <LF>, or both. Devices requiring nulls after <CR> are typically hard-copy devices requiring time for carriage movement. Devices requiring nulls after <LF> are typically CRTs which scroll slowly. Extra null characters are sent only in Converse and Command Modes.

PACLEN n Default: 80

Parameters: n = 0 - 255, specifying the maximum length of the data portion of a packet. The value 0 is equivalent to 256.

The TNC-220 will automatically transmit a packet when the number of input bytes for a packet reaches **n**. This value is used in both Converse and Transparent Modes.

If you perform file transfers, you should experiment with both **MAXFRAME** and **PACLEN**. If the link is good, there is an optimum relationship between the parameters set by these commands so that the maximum number of characters outstanding does not exceed the packet receive buffer space of the TNC receiving the data.

NOTE: Although there is no requirement for two TNCs exchanging data to have the same **PACLEN** value, allowing more than 128 characters of data in a packet may be incompatible with some varieties of TNCs.

PACTIME EVERY|AFTER n Default: AFTER 10

Parameters: n = 0 - 250, specifying 100 ms intervals.
EVERY - Packet timeout occurs every n*100 milliseconds.
AFTER - Packet timeout occurs when n*100 milliseconds elapse with no input from the computer or terminal.

This parameter is always used in Transparent Mode, and will also be used in Converse Mode if **CPACTIME** is **ON**. When **EVERY** is specified, input bytes are packaged and queued for transmission every **n***100 ms. When **AFTER** is specified, bytes are packaged when input from the terminal stops for **n***100 ms. In no case will a zero length packet be produced, and the timer is not started until the first byte is entered. A value of 0 for **n** is allowed, and causes packets to be generated with no wait time.

PARITY n Default: 3 (even)

Parameters: n = 0 - 3, selecting a parity option according to the following code: 0 = no parity, 1 = odd parity, 2 = no parity, 3 = even parity.

This command sets the parity mode for terminal or computer data transfer. The parity bit, if present, is automatically stripped on input and not checked in Command Mode and Converse Mode. In Transparent Mode, all eight bits, including parity if any, are transmitted in packets. If "no parity" is set and **AWLEN** is 7, the eighth bit will be set to 0 in Transparent Mode.

PASS n Default: $16 <CTRL-V>

Parameter: n = 0 - $7F, specifying an ASCII character code.

This command selects the ASCII character used for the "pass" input editing command. The parameter **n** is the ASCII code for the character you want to type in order to include the following character in a packet or text string. You can enter the code in either hex or decimal.

You can use this character to send any character in packets, even though that character may have some special function for the TNC-220. Suppose you are using the default Command Mode entry character <CTRL-C> and are accessing a Bulletin Board program that requires a <CTRL-C> to escape from some operation, you will type

<CTRL-V> <CTRL-C>

to insert a <CTRL-C> character in your packet. Of course, if you do this frequently you would be better off to change your Command Mode entry character.

A common use for the pass character is to allow <CR> to be included in the BTEXT and **CTEXT** messages. Similarly, you can include <CR> in text when you are in Converse Mode, to send multi-line packets. (The default send-packet character is <CR>.)

PASSALL ON|OFF Default: OFF

Parameters: ON - TNC-220 will accept packets with invalid CRCs.
OFF - TNC-220 will only accept packets with valid CRCs.

This command causes the TNC-220 to display packets received with invalid CRC fields. Packets are accepted for display despite CRC errors if they consist of an even multiple of 8 bits and up to 339 bytes. The TNC-220 will attempt to decode the address field and display the **callsign(s)** in the standard monitor format, followed by the text of the packet.

This mode is not normally enabled, since rejection of any packet with an invalid CRC field is what insures that received packet data is error-free. This mode might be enabled for testing a marginal RF link or during operation under other unusual circumstances.

If you set **PASSALL** ON and monitor a moderately noisy channel you will periodically see "packets" displayed in this mode, since there is no basis for distinguishing actual packets received with errors from random noise.

Logging of stations heard (for display by **MHEARD**) is disabled whenever **PASSALL** is ON, since the **callsigns** detected may be incorrect.

RECONNECT call#1 [VIA call#2[,call#3...,call#9]]

Parameters: call#1 - **Callsign** of TNC to be reconnected to.
call#2 - Optional **callsign(s)** of TNC(s) to be digipeated through.

RECONNECT is an immediate command. It may be used to change the path through which you are currently connected to a station. It may only be used when your TNC is connected on the current stream to the station to which you wish to **RECONNECT**.

The integrity of outstanding frames at the time of **RECONNECT** is not assured. For details regarding the parameter list, see the **CONNECT** command.

PORT n Default: 2

Parameters: n = 1 or 2, specifing a radio port

This command is used to change the radio port in use. Only the values 1 or 2 are accepted, corresponding to the TNC-220 radio ports 1 and 2.

If the **PORT** command is given without a parameter, the TNC-220 responds with the currently selected port.

If JP4 is installed, the **PORT** command causes the radio baud rate to be set to 300 when port 1 is selected, and 1200 when port 2 is selected.

This command should be used with caution since any link status will be transferred to the new port e.g. if a connection exists on one port and the **PORT** command is used to transfer to the other port without first completing a disconnection, the TNC-220 will attempt to maintain the connection without success out the newly selected port until the retry limit is exceeded. This practice will cause unnecessary radio transmissions. In addition, any stations digipeating through your TNC-220 on the first port will be left with an unuseable link situation.

REDISPLA n Default: $12 <CTRL-R>

Parameters: n = 0 - $7F, specifying an ASCII character code.

This command is used to change the redisplay-line input editing character. The parameter **n** is the ASCII code for the character you want to type in order to redisplay the current input line. You can enter the code in either hex or decimal.

You can type this character to cause the TNC-220 to retype a line you have begun. When you type the redispla y-line character, the following things happen: First, type-in flow control is temporarily released (if it was enabled). This displays any incoming packets that are pending. Then a <BACKSLASH> character is typed, and the line you have begun is retyped on the next line. If you have deleted and retyped any characters, only the final form of the line will be shown. You are now ready to continue typing where you left off.

You can use the redisplay-line character to see a "clean" copy of your input if you are using a printing terminal and you have deleted characters. If you have set **BKONDEL OFF**, deletions are designated with <BACKSLASH> characters, rather than by trying to correct the input line display. The redisplayed line will show the corrected text.

You can also use this character if you are typing a message in Converse Mode and a packet comes in. You can see the incoming message before you send your packet, without cancelling your input.

RESET

This is an immediate command. It resets all parameters to default settings and re-initializes the TNC-220. **WARNING:** All parameter customizing and monitor lists are lost.

If you want to re-initialize the TNC-220 using the parameter values in battery backed-up RAM, you should use the **RESTART** command or turn the TNC-220 off then on again rather than using this command.

RESPTIME n Default: 0

Parameters: n = 0 - 250, specifying 100 ms increments.

This command sets a minimum delay that is imposed on acknowledgment packets. This delay may run concurrently with default wait set by **DWAIT** and any random wait in effect.

This delay can be used to increase throughput during operations such as file transfer when the sending TNC-220 usually sends the maximum number of full-length packets. Occasionally, the sending TNC may not have a packet ready in time to prevent transmission from being stopped temporarily, with the result that the acknowledgment of earlier packets collides with the final packet of the series. If the receiving TNC sets **RESPTIME** to 10, say, these collisions will be avoided.

RESTART

RESTART is an immediate command. It re-initializes the TNC-220 using the defaults stored in bbRAM. The effect of this command is the same as turning the TNC-220 **OFF** then **ON** again. **RESTART** does not cause a reset of the parameters in bbRAM. See also the **RESET** command.

RETRY n   Default: 10

Parameter: n = 0 - 15, specifying the maximum number of packet retries.

The protocol allows for retries, i.e., retransmission of frames that are not acknowledged. Frames are re-transmitted **n** times before the operation is aborted. The time between retries is specified by the command **FRACK**. A value of 0 for **n** specifies an infinite number of retries. If the number of retries is exceeded, the TNC goes to the disconnected state (with an informative message if not in Transparent Mode). See also the **FRACK** command.

RXBLOCK ON|OFF   Default: OFF

Parameters: ON - The TNC will send data to the terminal in RXBLOCK format.
OFF - The TNC will send data to the terminal in standard format.

RXBLOCK is designed for automated operations, such as packet bulletin board stations. It is intended to help such systems discriminate between data received from the connected station and TNC-generated messages.

Correct operation of **RXBLOCK** is dependent on the AWLEN parameter getting set to 8 (bits) since the character FF hex marks the beginning of a recieved data unit header.

When RXBLOCK is on, data from other stations will be sent from the TNC in the following format:

$FF	L0	L1	PID	DATA
(Prefix)	(Length	)	(Pid)	(Data)

The fields above are defined as follows:

Prefix	$FF	::=	A character with all 8 bits set
Length	L0	::=	The high order data length, length and pid fields logically ORed with $F0
	L1	::=	The low order data length, length, and pid fields
Pid	PID	::=	The Protocol IDentifier byte received for the following data field
Data	DATA	::=	[Optional], variable length data

For best operation it is suggested that parameters like AUTOLF, MFILTER etc. be set OFF in order to prevent uncertanties in the size of the data field.

SCREENLN n   Default: 80

Parameters: n = 0 - 255, specifying the screen or platen width of the terminal in characters.

This command is used to properly format terminal output. A <CR> <LF> sequence is sent to the terminal in Command and Converse Modes after **n** characters have been sent. A value of zero inhibits this action.

If your computer automatically formats output lines, you should set **SCREENLN 0** to avoid a conflict.

SENDPAC n Default: $0D <CR>

Parameters: n = 0 - $7F, specifying an ASCII character code.

This command selects the character that will force a packet to be sent in Converse Mode. The parameter **n** is the ASCII code for the character you want to type in order to force your input to be packetized and queued for transmission. You can enter the code in either hex or decimal.

For ordinary conversation, you will probably set **SENDPAC $0D** and **CR ON**. This causes packets to be set at natural intervals, and causes the <CR> to be included in the packet.

If you have set **CPACTIME ON**, you will probably set **SENDPAC** to some value not ordinarily used (say, <CTRL-A>), and set **CR OFF**. This will allow you to force packets to be sent, but will not result in extra <CR> characters being transmitted in the text.

SOFTDCD ON|OFF Default: ON

Parameters: ON - The DCD signal uses the 8530 flag detection.
OFF - The DCD signal is obtained from hardware.

The TNC-220 can use several sources for the signal which serves as Data Carrier Detect (DCD).

When **SOFTDCD** is **ON**, the software obtains DCD indication from the flag detect feature of the 8530 SCC. Various software filters are used to reduce false DCD activations.

When **SOFTDCD** is **OFF**, the DCD signal is obtained from the 8530 DCD input on pin 19. This pin can be jumpered to either the AM7910 modem DCD signal, or the PLL lock signal generated on the optional tuning indicator.

The **SOFTDCD** ON command should be given when switching from port 1 (on HF) to Port 2 (on VHF).

START n Default: $11 <CTRL-Q>

Parameters: n = 0 - $7F, specifying an ASCII character code.

This command selects the User Restart character, which is used to restart output from the TNC to the terminal after it has been halted by typing the User Stop character. You can enter the code in either hex or decimal.

The User Stop character is set by the **STOP** command.

If the User Restart and User Stop characters are set to $00, software flow control to the TNC-220 is disabled, and the TNC-220 will only respond to hardware flow control (CTS).

If the same character is used for both the User Restart and User Stop characters the TNC-220 will alternately start and stop transmission upon receipt of the character.

STOP n Default: $13 <CTRL-S>

Parameters: n = 0 - $7F, specifying an ASCII character code. The code may be entered in either hex or decimal.

This command selects the User Stop character, which is used to stop output from the TNC-220 to the terminal. This is the character used to halt the TNC-220's terminal output and stop scrolling. Output is restarted with the User Restart character, which is set by the **START** command.

If the User Restart and User Stop characters are set to $00, software flow control to the TNC-220 is disabled, and the TNC-220 will only respond to hardware flow control (CTS). If the same character is used for both the User Restart and User Stop characters the TNC-220 will alternately start and stop transmission upon receipt of the character.

STREAMCA ON|OFF Default: OFF

Parameters: ON - Callsign of other station displayed.
OFF - Callsign of other station not displayed.

This command is used to enable the display of the connected-to station after the stream identifier. This is particularly useful when operating with multiple connections allowed. It is somewhat analogous to the use of **MRPT** to show digipeat paths when monitoring.

In the example below, the characters inserted by enabling STREAMCAll are shown in **bold face type**.

STREAMCAll <u>ON</u>

```
|A :K4NTA:hi howie
hello ted how goes it?
|B :WA7GXD:*** CONNECTED to WA7GXD
|B must be a dx record. ge lyle
|A unreal ted! fl-az no digis!
|B :WA7GXD:big band opening...ge
```

STREAMCAll <u>OFF</u>

```
|A hi howie
hello ted how goes it?
|B *** CONNECTED to WA7GXD
|B must be a dx record. ge lyle
|A unreal ted! fl-az no digis!
|B big band opening...ge
```

Thus, what looked like "|B" now appears as "|B:<callsign>:". This option is very useful for human operators trying to operate multiple simultaneous connections. It is probably less useful for "host" operations.

Note that, in the first example, the **STREAMSW**itch characters "|A" and "|B" with no ":" after them were entered by the operator of the TNC-220 to switch streams for his multiple-connect QSO(s). If you intend to operate multiple connections (as opposed to having your "host" computer operate multiple connections), use of this option is recommended.

<u>STREAMDBL</u> <u>ON|OFF</u> Default: OFF

Parameters: ON - Double all received **STREAMSW**itch characters.
OFF - Do not "double" received **STREAMSW**itch characters.

This command is used to display received **STREAMSW**itch characters by "doubling" them. The example below illustrates this action.

With **STREAMDBL** on, and **STREAMSW**itch set to "|", the following might be displayed from your TNC-220:

```
||this is a test.
```

In this case the sending station actually transmitted

```
|this is a test.
```

The same frame received with **STREAMDBL** OFF would be displayed as:

```
|this is a test.
```

When operating with multiple connections, this is useful for differentiating between **STREAMSW**itch characters received from other stations and **STREAMSW**itch characters internally generated by your TNC-220.

NOTE: The **STREAMSW**itch character must NOT be one of the letters A-J for this command to function properly.

<u>STREAMSW</u> n Default: $7C < | >

Parameters: n = 0 - $FF, usually specifying an ASCII character code.

This command selects the character used by both the TNC-220 and the user that a new "stream" (connection channel) is being addressed.

The character can be **PASS**ed in **CONVERS** mode. It is always ignored as a user-initiated stream switch in **TRANSPARENT** mode, and flows through as data. This means that the outgoing stream can not be changed while "on line" in **TRANSPARENT** mode (you must escape to **COMMAND** mode to switch streams).

For further usage of this character, see the **STREAMDBL** and **STREAMCA** commands.

TRACE ON|OFF   Default: OFF

Parameters: ON - Trace mode is enabled.
            OFF - Trace mode is disabled.

This command is used to enable the protocol debugging function. When **TRACE** is **ON**, all received frames will be displayed in their entirety, including all header information. In normal operation you will probably never need this function; however, if you need to report an apparent software bug, we may ask you to provide trace information if possible.

A trace display will appear in four columns on an 80-column display. Following is an example trace display. For comparison, the frame shown in the trace example would be monitored as follows:

```
KV7B>CQ,KF7B*:this is a test message
```

The trace display would look like this:

```
byte  ------------hex display------------shifted ASCII--  -----ASCII-----

000:  86A24040 40406096 AC6E8440 4060968C CQ     0KV7B  0KF  ..@@@@`..n.@@`..
010:  6E844040 E103F074 68697320 69732061 7B  P.x:449.49.0  n.@@...this is a
020:  20746573 74206D65 73736167 650D      .:29:.6299032.      test message.
```

The **byte** column shows the offset into the packet of the beginning byte of the line. The **hex display** column shows the next 16 bytes of the packet, exactly as received, in standard hex format. The **shifted ASCII** column attempts to decode the high order seven bits of each byte as an ASCII character code. The **ASCII** column attempts to decode the low order seven bits of each byte as an ASCII character code. In a standard AX.25 packet, the **callsign** address field will be displayed correctly in the **shifted ASCII** column. A text message will be displayed correctly in the **ASCII** column. Non-printing characters and control characters are displayed in both ASCII fields as ".". You can examine the hex display field to see the contents of the sub-station ID byte and the control bytes used by the protocol.

Protocol details are discussed in Chapter 9.

TRANS

This is an immediate command. It causes the TNC-220 to exit from Command Mode into Transparent Mode. The current link state is not affected.

Transparent Mode is primarily useful for computer communications. In this mode, the "human interface" features such as input editing capability, echoing of input characters, and type-in flow control are disabled. You may find Transparent Mode useful for computer Bulletin Board operations or for transferring non-text files. See discussion of Transparent Mode in Chapter 5.

TRFLOW ON|OFF   Default: OFF

Parameters: ON - Software flow control can be enabled for the computer or terminal Transparent Mode.
            OFF - Software flow control is disabled for the computer or terminal in Transparent Mode.

If **TRFLOW** is **ON**, the settings of **START** and **STOP** are used to determine the type of flow control used in Transparent Mode. If **TRFLOW** is **OFF**, only hardware flow control is available to the computer and all characters received by the TNC-220 are transmitted as data. If **START** and **STOP** are set to $00, disabling the User Stop and User Restart characters, hardware flow control must always be used by the computer.

If **TRFLOW** is **ON**, and **START** and **STOP** are non-zero, software flow control is enabled for the user's computer or terminal. The TNC-220 will respond to the User's Restart and User's Stop characters (set by **START** and **STOP**) while remaining transparent to all other characters from the terminal. Unless **TXFLOW** is also **ON**, only hardware flow control is available to the TNC-220 to control output from the terminal.

TRIES n Default: NONE

Parameters: n = 0-15, specifying the current **RETRY** level on the currently selected input stream.

This command is used to retrieve (or force) the count of "tries" on the currently selected input stream.

When used with no argument: if the TNC-220 has an outstanding unacknowledged frame, it will return the current number of tries; if the TNC-220 has no outstanding unacknowledged frames, it will return the number of tries required to obtain an acknowledgment for the previous frame.

If **RETRY** is set to 0, the value returned by issuing a **TRIES** command will always be 0.

This command is useful for obtaining statistics on the performance of a given path or channel. It should be especially useful for automatic optimizing os such parameters as **PACLEN** and **MAXFRAME** by computer-operated stations, such as automatic message forwarding stations using less-than-optimum paths (noisy HF or satellite channels, for example).

If followed by an argument, **TRIES** will force the "tries" counter to the entered value. This usage is **NOT** recommended.

TXDELAY n Default: 30

Parameters: n = 0 - 120, specifying 10 ms intervals.

This value tells the TNC-220 how long to wait after keying up the transmitter before sending data. Some startup time is required by all transmitters to put a signal on the air; some need more, some need less. In general, crystal controlled rigs with diode antenna switching don´t need much time, synthesized rigs need time for PLL lockup, and rigs with mechanical T/R relays will need time for physical relay movement. The correct value for a particular rig should be determined by experimentation. The proper setting of this value may also be affected by the requirements of the station you are communicating with.

Note that the TAPR TNC 1 and other TNCs using the same firmware interpret **n** in 40 ms intervals. The value of **TXDELAY** on TNC-220 will thus be 4 times the value used by a TNC 1 user to give the same delay time.

TXFLOW ON|OFF Default: OFF

Parameters: ON - Software flow control is enabled for the TNC-220 in Transparent Mode.
OFF - Software flow control is disabled for the TNC-220 in Transparent Mode.

If **TXFLOW** is ON, the setting of **XFLOW** is used to determine the type of flow control used in Transparent Mode. If **TXFLOW** is **OFF**, the TNC-220 will use only hardware flow control and all data sent to the terminal remains fully transparent.

If **TXFLOW** and **XFLOW** are **ON**, the TNC-220 will use the TNC Restart and TNC Stop characters (set by **XON** and **XOFF**) to control input from the terminal. Unless **TRFLOW** is also **ON**, only hardware flow control is available to the computer or terminal to control output from the TNC-220.

Note that if the TNC Restart and TNC Stop characters are set to $00, hardware flow control will always be selected regardless of the setting of **TXFLOW**.

UNPROTO **call#1** [VIA call#2[,**call#3...**,**call#9**]] Default: "CQ"

Parameters: call #1 - **Callsign** to be placed in the TO address field.
call #2 - 9 Optional digipeater call list, up to eight calls.

This command sets the digipeat and destination address fields of packets sent in the unconnected (unprotocol) mode. Unconnected packets are sent as unsequenced I frames with the destination and digipeat fields taken from call1 through call9 options. The default destination is for unconnected packets is CQ. Unconnected packets from other TNCs may be monitored by setting **MONITOR ON** and **BUDLIST** and **LCALLS** appropriately. The digipeater list is also used for BEACON packets (which are sent to destination address BEACON).

USERS n Default: 1

Parameters: n = 0 - 0, specifying the number of active connections that may be established to this TNC-220 by other TNCs.

USERS affects only the manner in which incoming connect requests are handled, and has no effect on the number or handling of connections you may initiate with this TNC-220. For example,

USERS 0 allows incoming connections on any free stream
USERS 1 allows incoming connections on stream A only
USERS 2 allows incoming connections on streams A & B
and so on through **USERS 10**.

XFLOWON|OFF Default: ON

Parameters: ON - XON/XOFF flow control is enabled.
OFF - XON/XOFF flow control is disabled and hardware flow control is enabled.

If **XFLOW** is **ON**, the computer or terminal is assumed to respond to the TNC Restart and TNC Stop characters set by **XON** and **XOFF**. If **XFLOW** is **OFF**, the TNC-220 will communicate flow control commands via RTS.

XMITOK ON|OFF Default: ON

Parameters: ON - Transmit functions are enabled.
OFF - Transmit functions are disabled.

When **XMITOK** is **OFF**, transmitting is inhibited. All other functions of the board remain the same, in other words, the TNC-220 generates and sends packets as requested, but does not key the radio PTT line.

You might use this command to insure that your TNC-220 does not transmit in your absence if you leave it operating to monitor packet activity. This command can also be used for testing using loopback or direct wire connections when PTT operation is not relevant.

XOFF n Default: $13 <CTRL-S>

Parameters: n = 0 to $7F, specifying an ASCII character code.

This command selects the TNC Stop character, which is sent by the TNC-220 to the computer or terminal to stop input from that device. You can enter the code in either hex or decimal.

This character would ordinarily be set to <CTRL-S> for computer data transfers. If you are operating your station in a Converse Mode and there is some chance that you might fill up the TNC-220's buffers, you might set this character to <CTRL-G> ($07), which rings a bell on many terminals.

XON n Default: $11 <CTRL-Q>

Parameters: n = 0 to $7F, specifying an ASCII character code.

This command selects the TNC Restart character, which is sent by the TNC-220 to the computer or terminal to restart input from that device. You can enter the code in either hex or decimal.

This character would ordinarily be set to <CTRL-Q> for computer data transfers. If you are operating your station in Converse Mode, and there is some chance that you might fill up the TNC-220's buffers, you might set this character to <CTRL-G> ($07), which rings a bell on many terminals.

MESSAGES

This section describes TNC-220 messages and the circumstances under which they appear.

|A

```
Pac-Comm Packet Radio Systems TNC-220
AX.25 Level 2 Version 2.0
Release n.n.n  mm/dd/yy - nnK RAM
Checksum $xx
cmd:
```

This is the sign-on message that appears when you turn on your TNC-220 or issue the RESET command. The release number indicates the software version. The checksum is a hex number for comparison with the checksum given for the firmware version you are using.

NOTE: The "|A" may display as a vertical-bar and any capital letter from "|A" through "J".

bbRAM loaded with defaults

This message appears along with the sign-on message above if the battery backed-up RAM checksum verification fails at power-on time, causing the TNC-220 to load the default parameters from ROM. (This will be the case the first time you turn on your TNC-220.) This message also appears if the TNC-220 loads the defaults in response to the RESET command.

cmd:

This is the Command Mode prompt. When this prompt appears, the TNC-220 is waiting for you to issue a command. Anything you type after this prompt will be interpreted as a command to the TNC-220. If a monitored packet has been displayed, the prompt may not be visible, even though you are in Command Mode. You can type the redisplay-line character (set by REDISPLA) to retype the prompt.

was

Whenever you change the setting of one of the TNC-220's parameters, the previous value will be displayed. This confirms that the TNC-220 properly interpreted your command, and reminds you of what you have done.

too many packets outstanding

This message would appear in response to a CONVERS or TRANS command, under special circumstances. If you have previously entered packet data filling the outgoing buffer in Converse Mode or Transparent Mode and then returned to Command Mode. You will be allowed to enter one of these modes when some of the packets have been successfully transmitted.

Command Mode Error Messages

If you make a mistake typing a command to the TNC-220, an error message will be printed. You may see any of the following messages depending on the type of error you have made.

?bad

You typed a command correctly, but the remainder of the line couldn't be interpreted.

?call

You entered a callsign argument that does not meet the TNC-220's requirements for callsigns. A callsign may be any string of numbers and letters, including at least one letter. Punctuation and spaces are not allowed. The sub-station ID, if given, must be a (decimal) number from 0 to 15, separated from the call by a hyphen.

?clock not set

The command **DAYTIME** was given to display the date and time without having previously set the clock. **DAYTIME** sets the clock if used with the daytime parameters, and displays the date and time if used alone.

?EH

The first word you typed is not a command or a command abbreviation.

?not enough

You didn't give enough arguments for a command that expects several parameters.

?not while connected

You attempted to change **MYCALL** or **AX25L2V2** while in a connected or connecting state.

?not while disconnected

You attempted to perform an operation that can only be done while connected. This message may appear when issuing a **RECONNECT** or **CONPERM** command.

?already connected to that station

The multi-connect software will not permit you to connect to the same station using multiple streams.

?range

A numeric argument for a command was too large.

?too long

You entered too long a command line, and the line was ignored. This might happen, for example, if you try to enter too long a message with **BTEXT** or **CTEXT**. If you get this message, the previous text entry was not changed.

?too many

You gave too many arguments for a command that expects several parameters. For example, **MFILTER** can have up to 4 arguments.

```
cmd:MFILTER $1B,$0C,$1A,$03,$07
?too many
```

?VIA

This message appears if you attempt to enter more than one callsign for the **CONNECT** or **UNPROTO** commands without the **VIA** keyword.

Link Status Messages

These messages inform you of the status of AX.25 connections your TNC-220 may be involved in. You can always interrogate the link status by giving the **CONNECT** command without parameters. If you attempt a connection when your TNC-220 is not in the disconnected state, the TNC-220 will display the link status but will take no other action. The following messages appear in response to the **CONNECT** command.

Link state is: CONNECTED to call#1 [VIA call#2[,call#3...,call#9]]

This display shows the station your TNC-220 is connected to and the digipeater route if any. The **callsign** sequence is the same sequence you would enter to initiate the connection.

*** LINK OUR OF ORDER, possible data loss [optional daytime stamp]

This message is issued upon failure of a CONPERMED link.

Link state is: DISCONNECTED

No connection currently exists. You may issue the CONNECT command to initiate a connection.

Link state is: CONNECT in progress

You have issued a connect request, but the acknowledgment from the other station has not been received. If you issue a DISCONNE command, the con nect process will be aborted.

Link state is: DISCONNECT in progress

You have issued a disconnect request, but the acknowledgment from the other station has not been received. If you issue a second DISCONNE command, the TNC-220 will go immediately to the disconnected state.

Link state is: FRMR in progress

The TNC-220 is connected but a protocol error has occurred. An improper implementation of the AX.25 protocol could cause this state to be entered. The TNC-220 will attempt to re-synchronize frame numbers with the TNC on the other end, although a disconnect may result. Connects are not legal in this state, and a disconnect will start the disconnect process.

The TNC-220 will inform you whenever the link status changes. The link status may change in response to a command you give the TNC-220 (CONNECT or DISCONNE), a connect or disconnect request packet from another station, a disconnect due to the retry count being exceeded, an automatic time-out disconnect (CHECK), or a protocol error.

*** CONNECTED to: call#1 [VIA call#2[,call#3...,call#9]]

This message appears when the TNC-220 goes from the "disconnected" or "connect in progress" state to the connected state. The connection may be a result of a CONNECT command you issued, or of a connect request packet received from another station.

*** connect request:call#1 [VIA call#2[,call#3...,call#9]]

This message indicates that the TNC-220 has received a connect request from another station which it has not accepted. This can happen if you have set CONOK OFF or if you are already connected to another station. When the TNC-220 types this message it also sends a DM packet (busy signal) to the station that initiated the connect request. If the TNC-220 rejects a connect request because you have set CONOK OFF, you can issue your own request to the station that called.

*** DISCONNECTED

This message is displayed whenever the TNC-220 goes to the disconnected state from any other link state. This message may be preceded by a message explaining the reason for the disconnect, below.

*** retry count exceeded
*** DISCONNECTED

This message is given if the disconnect was caused by a retry failure rather than by a disconnect request from one of the stations.

*** <callsign> busy
*** DISCONNECTED

This message indicates that your connect request was rejected by a DM packet (busy signal) from the other station. A Pac-Comm TNC-220 will reject a connect request if CONOK is OFF or if it is already connected to another station.

frmr frame just sent:
FRMR sent:string

The TNC-220 is connected, and a protocol error has occurred. The TNC-220 has sent a special FRMR packet to attempt to re-synchronize frame numbers with the TNC on the other end. The string is replaced with the hex codes for the three bytes sent in the information part of the FRMR frame. This message will not appear if your TNC-220 is in Transparent Mode.

FRMR rcvd:

This message is followed by a display of the FRMR packet received in the trace display format. This format is explained in the TRACE command entry. This message will not appear if your TNC-220 is in Transparent Mode.

Chapter 7

HARDWARE

This chapter includes detailed hardware specifications and a functional description of the hardware design of the Pac-Comm TNC-220.

TNC-220 Specifications

Processor Z-80 (8400)

Clocks Processor master clock input frequency: 2.4576 MHz jumper selectable to 4.9152 MHz (higher clock rate may require use of Z80B CPU, 6 Mhz 8530 and 200 nSec EPROMs).

Memory All memory in industry-standard JEDEC Byte-Wide sockets.

Standard complement of ROM: 32k = 1 x 27256

Standard complement of RAM: 16k (2 x 6264LP). RAM can be expanded to 32k by substituting one 62256 in memory socket 1 (socket 2 empty).

Serial Z80 SCC (8530) port B configured as a UART plus standard 1488 and 1489 RS-232C signal level interface. TTL signals are selected and buffered through a 74HC157 selector chip.

Baud rates supported: 300, 1200, 2400, 4800 and 9600, assignable in software. Cold-start baud rate is 1200.

Standard female DB-25S (DCE) RS-232C connector. Normally unused secondary RS-232 pins on this connector support TTL level serial signals when TNC-220 is jumpered for TTL signals.

Modem Built-in large scale integrated circuit modem supports conventionally used tones for HF and VHF operation as well as 400 Hz shift for experimentation. The modem requires no calibration.

The modem is configured with dual radio connectors, filters, and radio keying circuitry. The HF filter is a 6 pole active filter with a preamplification stage. The VHF filter is a two stage preamplifier and low-pass filter. Each filter may be bypassed if desired.

Port 1 is equipped with a separate Frequency Shift Keying (FSK) circuit for direct keying of FSK compatible transceivers.

An external modem may be attached via a single connector which completely bypasses the on-board modem.

Support for an optional internal tuning indicator is provided.

bbRAM Non-volatile storage of all important operating parameters is accomplished by using a battery backed-up system for the entire 16k (or 32k) bytes of system RAM. Battery backed RAM scanning checks all ten possible connection control structures.

Protocol AX.25 Level 2 is supported. Pre-Version 2.0 support is compatible with earlier TAPR TNC-1 style units running 3.x software. Full support of Version 2.0 protocol is provided. Full duplex radio link operation is supported.

Operating Modes Command Mode: accepts commands via user port.

Converse Mode: accepts digital data, transmits and receives packets, permits terminal editing features (character delete, line delete, input packet delete, output packet delete and redisplay input) via special characters trapped by the TNC-220. Escape to command mode via special character or BREAK signal. Optional use of packet completion timer as in Transparent mode.

Transparent Mode: accepts digital data, transmits packets via packet completion timer or buffer full only, and receives packets. No local editing features permitted. Escape to command mode via specially timed character sequence or BREAK signal.

Power +12 to +16 volts DC at less than 600 mA.

General Description

The TNC-220 is based on the Zilog Z80 (tm) family of microprocessor components. All parts used in the TNC-220 should be readily available to the Radio Amateur. The design is not 'TNC-2 compatible' and will not operate with software (EPROMS) designed for that series of units.

Major electronic devices in the TNC-220 include a CPU (Central Processing Unit) for controlling the TNC-220 and an SCC (Synchronous Communications Controller) for providing serial interface ports to the on-board modem (which connects to your radio) and to the RS-232C/TTL serial terminal port (which connects to your computer or terminal). The modem is the AMD AM7910 large scale integrated component.

The TNC-220 also includes two types of memory. ROM (Read Only Memory) stores the program that tells the TNC-220 how to implement the AX.25 protocol. Battery backed-up RAM (Random Access Memory) provides a scratch-pad area for temporary data as well as non-volatile storage for operating parameters such as your station call sign. The battery back-up feature enables the TNC-220 to retain these values when power is off so they don't have to be entered every time the unit is used.

Other integrated circuits are used for functions including clock oscillator, clock signal divider, memory address decoder, power supply and voltage inverter, transmit watch-dog timer and modem filter. Refer to the schematic diagram while reading the following circuit descriptions.

Detailed Circuit Description

Oscillator

The crystal, portions of the 74HC04, and associated resistors and capacitors form a crystal oscillator circuit to provide a stable clock signal for system timing.

Dividers and Baud-rate Generator

The 74HC4020 provides clock outputs derived from the oscillator which provide both normal and high-speed clock signals for the CPU, SCC and modem, a timing strobe for the time-of-day clock, and baud rate generation for full duplex operation.

CPU, SCC and associated circuitry may be operated at 2.4576 MHz (standard) or 4.9152 Mhz via the JMP-S jumper. Operation at the faster speed may require higher-speed parts.

The output from counter 74HC4020 pin 3 provides a real-time clock interval signal for the SCC. During normal operation, the SCC will be programmed to interrupt the CPU on every transition of this 150 Hz signal. This interrupt occurs 300 times a second, and is used for protocol and timing functions.

CPU Complex

The 27256 EPROM provides system ROM for program storage. The 74HC139 and 74HC164 act as a memory address decoder, mapping the ROM into the CPU's memory address space beginning at address 0, and along with two jumpers, allows selection of either 16k or 32k of RAM.

Static RAMs provide system RAM for temporary scratch-pad storage, message buffers, etc. Also, because the RAM is backed up by a battery and will not lose its contents when the main power is removed, it is used to provide semi-permanent storage of user-supplied information (such as your call sign). RAM starts at address 8000 hex. Two jumpers labelled A B allow selection of 16k bytes (both jumpers set to A) using two 8k RAMs or 32k bytes (both jumpers set to B) using one 32k RAM.

Diodes in the power supply circuitry ensure that the RAM is in the "power-down" state for minimum battery power consumption whenever power is removed from the TNC-220.

Serial Interface

The Serial Communications Controller (SCC) provides two channels of serial I/O.

The SCC B channel is used for the computer or terminal interface. The 1488 serves as RS-232C drivers while the 1489 serves as RS-232C receivers. The 74HC157 serves to select and buffer the signals output to a TTL computer.

The SCC channel A is used for the radio/modem interface and is normally operated as a full duplex HDLC channel for compatibility with the AX.25 protocol specification.

Watch-dog Timer

The LM555 programmable timer provides a "watch-dog" timer on the transmit key line (PTT). If the PTT is asserted for more than approximately 45 seconds, the CPU and SCC are reset and the software is reinitialized. This prevents "locking-up" a packet channel and also ensures compliance with FCC regulations regarding unattended station operation.

No watch-dog disabling jumper is provided for testing purposes since no modem calibration is ever required. When transmitter drive level adjustments are being performed using TNC-220 transmitted signals, the TNC-220 should be unkeyed approximately every 25 seconds to reset the timer. If long duration continuous transmission is required for some reason, the 555 timer chip (near the battery) may be removed from its socket to disable the circuit.

Modem

The 7910/7911 integrated circuit modem employs a modulator that generates an audio data signal for use by the radio transmitter. Two tones are used, one for each digital level. Tone selection is provided by modem configuration jumpers.

Dual radio ports are provided to allow the permanent connection of two radio transceivers. The jumpers allow separate tone pairs to be automatically selected by software for each of the dual radio ports.

Each radio port is equipped with an active filter to optimize that port's performance for the appropriate tones (Port 1 HF, Port 2 VHF). The HF filter is a preamplifier/limiter followed by a six pole bandpass filter. The VHF filter is a preamplifier/limiter followed by a 2 pole low pass filter. Either filter may be bypassed with appropriate jumper settings. Provision is made for an off-board HF filter on the second port.

A 10k ohm trimpot is used on each port to set the output level to the transmitter.

A 74HC4053 analog switch and associated circuitry is used to allow software selection of either radio port, including selecting the appropriate filter, tone pair, and baud rate.

Power Supply

The regulator (7805) and associated components provide a +5 volt regulated output for the TNC-220 digital logic circuitry. The on-board battery and control circuitry provides a "power failure" back up for the RAM chip which contain operating parameters to ensure that RAM is in the "power-down" state when the main power is removed. Diodes are used to isolate the battery from the +5 volt line when main power is available. JMP-B provides a means of disconnecting the lithium battery for TNC-220 maintenance.

The LM555 and is charge-pump voltage inverter which generates a regulated -10 volts for the RS-232C drivers and -5 volts for the 1458 op-amp filters and the modem.

Jumper Functions

The following table lists the function of each jumper on the TNC-220 PC board. The default (pre-wired) or normal (standard jumper setting) positions for each are shown.

Table 7-1. TNC-220 Jumper Functions.

Jumper	Position	Function
JP-S	L (normal)	2.46 MHz CPU clock
	H	4.92 MHz CPU clock
JP-B	ON (normal)	Lithium battery connected
	OFF	Lithium battery disconnected
Two Jumpers, each labeled "A B"	A (16k version)	Two 8k RAMs installed
	B (32k version)	One 32k RAM installed in -1 position

NOTE: Both jumpers must be in the same (either A or B) position.

Jumper	Position	Function
JP-T	232	RS-232C serial port enabled
	TTL	TTL serial port enabled
JP-F	IN (normal)	HF filter selected
	OUT	HF filter bypassed
JP-V	IN (normal)	VHF filter selected
	OFF	VHF filter bypassed

NOTE: External audio signals may be input into the ´IN´ pin and nearby ground pad.

Jumper	Position	Function
JP1	1 & 2 (normal)	Required (with JP4 ON) for software control. Both ports use Bell 202 tones (JP2 & JP4 OFF)
	2 & 3	Both ports use Bell 103 Originate tones (JP2 & JP4 OFF). Both ports use CCITT V.21 tones (JP3 & JP4 OFF)
JP2	ON (normal)	Required (with JP4 ON) for software control
	OFF	
JP3	ON (normal)	Required (with JP4 ON) for software control
	OFF	
JP4	ON (normal)	Software tone selection (JP1 required to be set at 1 & 2, JP2 & JP3 both ON.) Port 1: Bell 103 Originate tones, Port 2: Bell 202 tones.
	OFF	Manual tone selection by JP1/JP2/JP3
JP-L	NORM (normal)	Normal modem operation
	LOOP	Digital loopback mode
JP-D	M (normal)	Hardware DCD obtained from AM7910 modem.
	T (tuning ind)	Hardware DCD obtained from optional tuning indicator.

NOTE: JP-D must be in the M position if no tuning indicator is installed.

Jumper	Position	Function
JP-C	HIGH (normal)	DB-25 pin 8 pulled up to +10 volts.
	CONN	DB-25 pin 8 reflects status of CON LED.

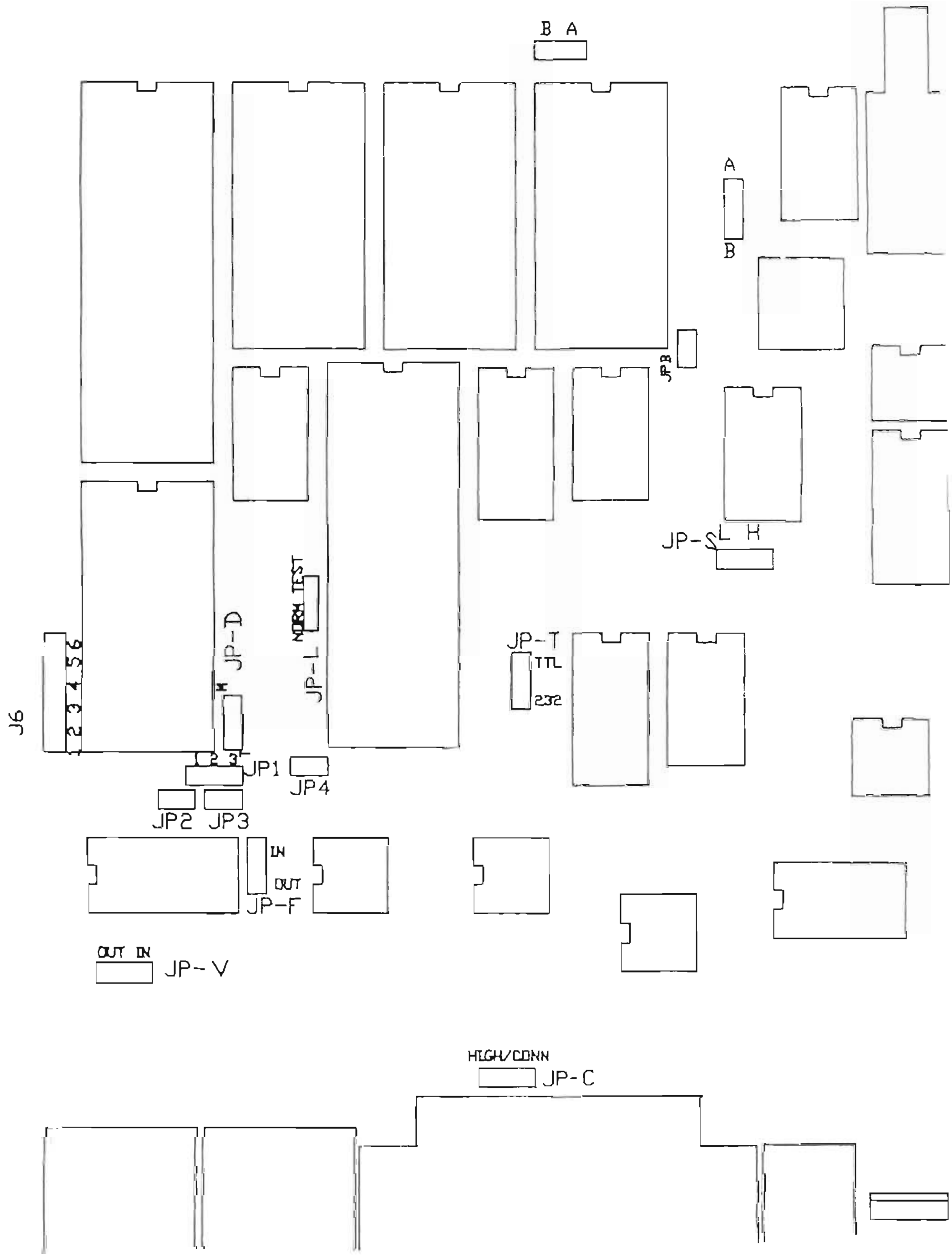


Figure 7-1. TNC-220 Jumper Locations

RS-232C Handshaking Protocol

The CTS, DSR and DTR lines of the RS-232C port (J3) are used for hardware "handshaking" protocol to control the flow of data between the terminal (DTE) and the TNC-220 (DCE).

The TNC-220 always asserts (makes true) Data Set Ready (DSR) on J3 pin 6 via a pullup resistor. Thus, whenever the TNC-220 is powered up, it signals to the terminal connected to J3 that the TNC-220 is "on line."

The terminal indicates it is ready to receive data from the TNC-220 by asserting its Data Terminal Ready (DTR) output, J3 pin 20 (also connected to J3 pin 4). The TNC-220 will send data when it has data to send and DTR is asserted. If the terminal is not ready to receive data, it should negate (make false) DTR to the TNC-220. Thus, data flow from the TNC-220 to the terminal is controlled by the use of the DTR line. The state of the DTR line is ignored by the software if "software flow control" is enabled in this direction.

The TNC-220 asserts its Clear To Send (CTS) output, J3 pin 5, whenever it is ready to receive data from the terminal. If the TNC-220's buffers fill, it will negate CTS, signaling the terminal to stop sending data. The TNC-220 will assert CTS when it is again ready to receive data from the terminal. Thus, data flow from the terminal to the TNC-220 is controlled by the use of the CTS line. The CTS line is always asserted if "software flow control" is enabled in this direction.

Some serial I/O ports do not implement CTS, DTR and DSR handshaking. If these pins are not connected at the terminal end, they will be pulled up (and thus asserted) by resistors at the TNC-220 end. However, a non-standard serial connector may use some pins for other purposes, such as supplying power to a peripheral device, so be sure that your system either implements the CTS DTR and DSR handshake or has no connections to these pins of J3 whatsoever. Note that reference to RS-232C "compatibility" or the presence of a DB-25 type connector does not guarantee that you have a full RS-232C serial port!

NOTE Use of a ´25 wire´ RS-232 cable is not recommended unless the user can determine that no interfering signal/voltage levels are present on pins 12, 13, 14, 16, or 19.

The Pac-Comm TNC-220 supports asynchronous terminal baud rates of 300, 1200, 2400, 4800 and 9600. The port supports standard parity options as well as 7- or 8-bit character lengths. Setting these terminal parameters is discussed in Chapter 4.

If you want to interface your TNC-220 with a device configured as DCE, such as a telephone modem or another TNC, a so-called "null modem" cable may be constructed to interchange the data and handshake signals.

Modem Disconnect - J5

The modem disconnect on the TNC-220 PC board (J5) is provided for using an external modem with the TNC-220. This allows use of higher-speed modems, or more sophisticated, higher-performance modems for satellite or other uses.

The following information is primarily for those who wish to interface external modems to the TNC-220. Familiarity with modem and serial data channel terms is assumed.

A physical connector for J5 is not supplied with the TNC-220. Any standard 20-pin header for use with IDC cable connectors should work. Suitable parts are the Ansley 609-2027, the 3M 3428-6202, etc.

When installing the connector, be sure to line up the marked pin (pin 1) of the header with the the PC board pin 1 marking. Cut the traces between pins 1-2, 3-4, 5-6, 9-10, and 19-20 on the top of the TNC-220 PC board with a sharp knife or MotoTool. CAUTION: DO NOT CUT ANY OTHER TRACES WHICH MAY PASS THROUGH THE J5 AREA AND WHICH ARE NOT CONNECTED TO J5 PINS! Once a connector is installed at J5, you must install push-on jumpers or a shorting connector on the above pin pairs to use the on-board modem.

The signals used at connector J5 are at standard TTL interface levels. A TTL high, or 1, is greater than +2.4 volts but less than +5.25 volts. A TTL low, or 0, is less than 0.8 volts but greater than -0.4 volts. **DO NOT connect an RS-232C level modem directly to J5!**

NOTE: The modem disconnect is not identical to that used in TAPR TNC 1 or TNC-2. Carefully consult the circuit schematic and J5 pin definitions before interfacing an external modem using the same cabling you may have prepared for use with TAPR TNC 1 or 2.

Pin 1 Carrier Detect Input

This pin tells the 8530 radio port that a valid data carrier has been detected. It should be pulled high when no carrier is detected and low when a carrier is present. It is not used by the on board modem or software.

Pin 2 Not used

Pin 3 8530 Received Data Input

This pin is the received data input to the 8530. This pin is normally jumpered to pin 4 when the on-board modem is used.

Pin 4 Modem Data Output

This signal is the digital data output from the on-board modem. It is normally jumpered to pin 3 when the on-board modem is used.

Pin 5 8530 RTS Output

This signal is used for transmitter activation. It activates the PTT pin of the radio connector via the watch-dog timer.The 8530 will pull this output low when the TNC-220 wants to transmit; otherwise it will remain high. This pin is normally jumpered to pin 6 when the on-board modem is used.

Pin 6 Modem RTS Input

This signal is an input to the on-board modem. It should be left high and pulled low only when transmission is desired. This pin is normally jumpered to pin 5 when the on-board modem is used.

Pin 7 Not Used

Pin 8 Not Used

Pin 9 8530 CTS Input

This pin is an input to the 8530. It is high when the attached modem is not ready to accept data, and low when the attached modem is ready to accept data. The TNC-220 will not attempt to send data when this pin is high. This pin is normally jumpered to pin 10 when the on-board modem is used.

Pin 10 Modem CTS Output

This pin is used by the on-board modem to signal the 8530 when it is ready to accept data. This pin is normally jumpered to pin 9.

Pin 11 Not Used

Pin 12 Not Used

Pin 13 Not Used

Pin 14 Not Used

Pin 15 TNC-220 Ground Reference

This pin ties to the TNC-220 digital ground system, at the 8530.

Pin 16 Not Used

Pin 17 8530 SYNCA Pin

This pin ties to the 8530 SYNCA pin which is a programmable multi-purpose input/output.

Pin 18 Not Used

Pin 19 Transmit Data Output

This line is the 8530 data output. This pin is normally jumpered to pin 20 when the on-board modem is used.

Pin 20 Transmit Data Input

This input line accepts data to be be transmitted by the modem. This pin is normally jumpered to pin 19 when the on-board modem is used.

If you elect to use an off-board modem, be sure to properly shield the interconnecting cables for RFI protection.

Tuning Indicator Interface - J6

In order to facilitate communications on HF and OSCAR, the Pac-Comm TNC-220 includes a connector for attaching a tuning indicator. A mating Tuning Indicator mounts in the TNC-220 cabinet. The indicator is documented separately.

The connector pinouts are as follows.

Pin 1 Data Carrier Detect (DCD) input to the 8530 port A (pin 19) via the ´ ´ postion of JP-D.

Pin 2 Analog modem input signal (received signal) for use by the tuning indicator.

Pin 3 Analog modem output signal for tuning indicator calibration purposes.

Pin 4 +5 volts at up to 100 mA.

Pin 5 -5 volts at up to 10 mA.

Pin 6 Ground

The Tuning Indicator option for the TNC-220 is built on a separate printed circuit board which fits into the upper set of rails in the TNC-220 cabinet. The tuning indicator accepts the received packet signal in parallel with the 7910 modem via the connection at J6 pin 2. An XR2211 PLL demodulator generates an error voltage when a packet signal is received. The error voltage is used to drive a 20 segment LED bargraph display which provides tuning information to the packet operator. The 2211 also generates a lock signal which is transmitted to the TNC-220 via J6 pin 1 for use as a DCD signal when selected by the TNC software.

TROUBLESHOOTING

WARNING: Never remove or insert an IC with power on!

Your Pac-Comm TNC-220 is a complex piece of electronic equipment. Servicing must be approached in a logical manner. The best preparation for troubleshooting is to study the detailed hardware description in Chapter 7. While it is not possible to present all possible problems, symptoms and probable cures, this section of the manual will give direction to troubleshooting based on our experience.

General Tests

When things go wrong... the first place to start is with your visual inspection. If you built the TNC-220 as a kit, look at the following:

1. Parts (ICs, transistors and diodes) in backwards, or in feedthru holes.
2. Solder bridges or shorts caused by bent over leads. This, in our experience, causes over 80 percent of new failures.
3. Burned or discolored components (or components that will burn you--except the power resistor, it does get hot).

If this doesn't resolve the problem, it is a rare failure. If an oscilloscope is not available, then a voltmeter can help you find serious difficulty. Set your meter for the scale where a full 5 volts can be displayed. Check for a reading of approximately 4 volts at the clock locations of Z80-6, 8530-20, 7910-24, and 74HC164-8. This should indicate a good clock circuit. A reading of over 4.5 volts or under 2.5 volts is most likely an indication of trouble. Next, check Z80 pin 26 in that it does in fact go low while you ground the external reset line. It should normally be high to let the processor run. The next step involves checking all the data and and address lines on the Z80, 8530, 27256, and the RAM chip (6264, or 62256). Look for a moderate voltage, somewhere around 2.5 to 3.5 volts. A very low, or very high voltage reading is a sign of no activity, or open and shorted circuits. If you cannot pin-point the trouble using this set of guidelines, then you must either obtain an oscilloscope, or send your board back to us for repair.

If you have an oscilloscope available, the trouble shooting is much easier! The checks made above still apply, but now you can "see" trouble on the oscilloscope waveforms. If you see a 'stair-stepped' waveform on two different pins, then it is likely they are shorted together. Also, square waves of half amplitude, or lines with no activity are dead giveaways on the scope. The scope also allows you to trace the analog sections of the device, including the modem.

Component failure from the factory is rare, but failure may be induced by a short on the circuit board. If you suspect a component failure, contact us for any replacement part. Contact us at our tech help line, but please go through all the above steps before you call us, to speed our efforts to get you up and running.

Be very careful about shorting pins on ICs when applying meter or scope probes to the board. It is a good idea to attach a secure ground lead to the meter or scope, one that won't accidentally short across components on the board. A good place to pick up this ground is on the ground bus that completely encircles the perimeter of the board. Avoid connecting in the area where power leads run to and from the power switch to avoid the possibility of a short.

WARNING: IF THE TNC-220 IS OPERATED OUTSIDE THE CABINET FOR EVEN A FEW SECONDS, THE 7805 VOLTAGE REGULATOR MUST BE PROPERLY HEATSINKED TO PREVENT FAILURE!! The 7805 regulator must dissipate up to 3.5 watts of electrical power as heat. This is well within the normal ratings for the device IF properly heatsinked. Normally the 7805 is firmly connected to the TNC-220 cabinet for heat transfer. Use the paper clasp provided for a temporary heatsink. It is suggested that a large screwdriver blade be slipped through the 'handles' of the paper clasp to augment the heat dissipation for periods beyond approximately 5 minutes.

Step 1: Power Supply

The first thing to check in any malfunction is the power supply. Check the power supply levels at the outputs of the voltage regulators as well as the output of the inverter 555. Are they close to their nominal values? Do all the ICs in the suspected area have the proper voltage on their power pins? Is there excessive ripple in any of the DC voltage lines? If so, check the regulator and associated components, working backwards toward the input power switch. If the voltage is low, in conjunction with a hot regulator, suspect a short circuit on the board.

If the problem is in the -5 volt supply, work backwards through the negative voltage generator. The junction of the 68 ohm resistor and 1N751, which should be -5 volts regulated, and the junction of the 68 ohm resistor and 1N914 should be approximately -10 volts unregulated. If the 1N914 test point is more negative than -7 volts (i.e., normal), but the -5 volt regulated voltage is wrong, check the negative regulator components. If both the -10 and the -5 volt points are wrong, look for shorts. If no voltage appears at pin 3 of the 555, but proper supply voltage appears on pin 8, the 555 may be at fault. Verify that the 555 is oscillating by looking at the wave form at pin 2.

Step 2: Obvious Problems

Look for any unusual physical symptoms. Have you installed any ICs the wrong way? Are any components discolored? Does something smell burnt? Do any of the parts seems excessively warm? In general their normal temperature will be well below the boiling point of water, but you may not want to keep your finger on them very long.

Step 3: Assembly Problems

Carefully inspect the PC board and component installation. Are any cold solder joints present? (See the soldering instructions in the **TNC-220 Assembly Manual**.) Is a metal screw shorting to the board anywhere? Are all ICs firmly seated in their sockets? Are any IC leads tucked under the chip or otherwise bent in such a manner that they aren't making proper contact with the IC socket? (This is a **very** common error.)

Inspect the diodes and electrolytic capacitors for proper installation. Are the diode cathodes pointing the correct way? Are the negative ends of the electrolytic capacitors pointing the correct way?

Step 4: Cabling Problems

Inspect the interconnection cabling. Does it work on another TNC? Has the radio and/or terminal been successfully used on packet with this or another TNC? Are all the connections tight? Has the cable frayed or broken?

Specific Symptoms

While the steps described above may seem obvious, careful inspection often will point to the problem or give significant clues as to the probable area of the TNC most suspect. After the above inspection has been completed and apparent problems dealt with, it is time to proceed to more specific analysis.

Symptom: TNC appears dead

If the TNC powers up with the PWR and DCD LEDs lit, followed by the DCD extinguishing a second or so later, the processor is working and the software is probably working correctly. You should suspect the terminal port at this point. Check all connections and verify the logic levels according to the terminal interface troubleshooting section in this chapter.

Oscillator and Reset Circuits

If no LEDs wink during the reset cycle the problem is more serious. Check to see that the crystal oscillator is working (4.1952 MHz) and that the signal appears on the input (pin 10) of the divider IC, 74HC4020. Confirm that JP-S is in the "L" position unless higher speed CPU and 8530 parts have been installed. The clock signal is a (possibly distorted) square wave signal. Check for clock signals (2.4576 MHz) at 4020 pin 9 and on Z80 pin 6, 8530 pin 20, 7910 pin 24 and 74HC164 pin 8.

Verify that the battery backed-up RAM protection circuit, is working by checking that pin 28 of RAM-1 is going to at least -4.7 volts after input power is applied to enable normal operation of the static RAM chips.

Digital Logic Lines

Remember that all the logic circuits operate at standard TTL levels (a "low" is less than +0.8 V and a "high" is greater than +2.4 volts), and all digital inputs and outputs switch between these two levels. Thus, if you see logic signals switching between 0 and, say, 1 volt, you can be sure there is a problem (usually a short). On the other hand, do not mistake switching transients on digital logic lines for improper operation -- these show up as ringing and other distortions.

Verify that there is activity on the control bus READ and WRITE lines, the CHIP-ENABLE lines on the memories (pins 20), the IOREQ line on Z80 pin 20 and the INT line on Z80 pin 16. Each of these lines should show activity, and if any line is quiet this is a sign of trouble. Address and data line problems may also show up as lack of activity on the control bus lines, especially the chip selects. Check each of the 16 address and 8 data lines for activity. Any lines showing a lack of activity are not operating properly.

If you suspect problems with address or data lines, try removing all the memory chips. Each address and data line will now show a distinct pattern. The address lines should be (possibly distorted) square waves whose periods increase by a factor of two on successive lines as you step line by line from A0 to A15.

Logic lines that show no activity may often be traced to a short on the pc board, probably due to a solder splash or bridge. If you decide to use an ohmmeter to check for shorted lines, use a low voltage/low current test instrument. (Most modern DVMs are fine for this.) If in doubt, remove any ICs connected to the lines you are measuring. If you suspect a short, check the high density areas of the PC board for the problem. In most cases the short will be found there. It is very unlikely that the PC board itself will have a short. An overall check of the processor, memory, and 8530 may be conducted using the **HEALTH** command (see Chap 6.)

Symptom: Modem won't key transmitter.

NOTE: Be sure you have commanded the software to select the port you are attempting to use. If the transmitter doesn't key, the problem is most likely in the connecting cable. If your connections appear to be in order, check 8530 pin 17 for keying (PTT) transitions. If nothing is seen on pin 17, either software or digital hardware are faulty. If pin 17 is active, trace the signal through the 2N2222 inverting transistor, the 4053 switch to the gate of the appropriate keying FET. The FET output line changes from high impedance to low when a keying signal is present so do not expect to see activity on it unless a radio is attached.

Symptom: Uncopyable transmitted or received packets

If no one seems able to decode your packet transmissions, it is often the case that your transmitter is being overdriven, or that the modem configuration jumpers JP1, JP2, JP3, or JP4 are not in the proper postions. You can confirm that the digital logic and modem on the TNC-220 are operating properly by performing an Analog Loopback check [Select **PORT** 2, **FULLDUP ON**, and install a jumper between J2 pins 1 and 4]. If the loopback works properly, the solution is to reduce the drive level via the trimpot for the port in use. Note that direct connection to typical microphone inputs requires the drive trimpot to be turned to near the maximum signal position to produce sufficient signal levels. If the Analog Loopback fails to operate properly, move the JP-L jumper to 'LOOP' and perform a digital loopback. If this test operates correctly, the CPU, software, and 8530 are operating correctly and the modem IC is suspect.

Symptom: DCD LED constantly flickers on and off

If the DCD LED flickers with no signal cable connected to the TNC, check the negative 5 volt supply for ripple and noise. A defective, or incorrectly installed filter capacitor is indicated. If the flickering only occurs with the signal cable connected, be sure the radio is operating with squelch closed, that shielded signal cable is being used, and that the audio drive level from the radio is not excessive. If these areas all check out, consider reducing the value of the gain resistor on the op amp for the affected port, R4 or R21.

Symptom: Transmitter locks in key down condition

The TNC-220 is protected against PTT malfunction by a watchdog timer on the keying line. However, if unshielded, improperly grounded cabling is used to feed the transmiter key line, and other signal cables, RF energy can get back into the TNC and lock up the keying transistor. Precautions are especially necessary if a high power amplifier is being used and there is ´lots of RF´ in the area of the TNC.

Terminal Interface Troubleshooting

If you can´t get the TNC to sign on and accept data from your terminal or computer, the problem may be in the serial interface. NOTE: The primary cause of non-communication between the TNC and terminal is improper cabling setup.

Symptom: TNC won´t sign on to the terminal.

If you find the TNC won´t send data to your terminal, verify that the DTR line at pin 4/20 of J3 is not being held low. If the software flow control option is disabled, the TNC will not send data to the terminal unless its DTR is asserted. If the terminal does not implement the DTR/CTS protocol, the DTR/CTS lines (pins 20/4 and 5 on J1) should remain unconnected.

If the above checks are ok, observe pin 25 of the 8530 with an oscilloscope and cycle the TNC power switch. Transitions on this pin shortly after reset indicate that the TNC is sending data. If no signal transitions are seen, recheck that the CPU appears to be operating (see above.) If data appears to be present, verify that TTL transitions are also present on the 1488 pin 4/5 and RS-232 signal transitions on 1488 pin 6 and J3 pin 3.

Symptom: The TNC appears to be signing on but prints only gibberish.

This indicates that some combination of the data rate (baud rate), parity option, or number of start and stop bits are not set the same at the TNC and at the terminal. If possible, set your terminal to 1200 baud. Also verify that the terminal is set for seven data bits, space parity, and 1 stop bit. These are the default sign-on settings stored in EPROM. Perform a hard reset by the power switch OFF then ON (out then in). The sign on message should appear.

Symptom: The TNC signs on OK but won´t accept commands.

If you are using a TTL interface, be sure that JP-L is in the TTL position. The TNC-220 will send sign-on characters out the TTL port even if JP-L is in the RS-232 position, but will not accept any commands from the terminal.

After the TNC signs on, try giving it any command. If the default settings are in effect, it will attempt to echo each character you type back to the screen. If it doesn´t echo, be sure that the SCC pin 22 has a voltage level between 0 and +0.8 volts on it. The voltage on 1489 pin 10 should be greater than +3 volts.

If the above checks are OK, use an oscilloscope to verify that data is present on 8530 pin 27 and 1489 pin 1 when you strike a key on your terminal. If not, the data isn´t getting from your terminal to the TNC. Check J3, the cable and 1489 again. Finally, be sure that your terminal actually uses levels less than -3 volts and greater than +3 volts for signal levels. 0 and +5 volts will not work unless the TNC-220 is jumpered for TTL operation.

Chapter 9

PACKET RADIO PROTOCOL

Explanation of Protocol

The material in this chapter is intended to supply an overview of the protocol used to transmit data by the Pac-Comm software. References are given to more detailed information required by those wishing to implement these protocols on other hardware. The material presented below is somewhat tutorial in nature for those who have not had previous exposure to layered network protocols, but it presumes some knowledge of general communications hardware and software. Persons already well versed in networking may want to skip this chapter and refer to the primary defining document, Amateur Packet-Radio Link-Layer Protocol, AX.25 Version 2.0, available from the ARRL, 225 Main Street, Newington, CT 06111.

The Pac-Comm TNC hardware and software architecture is organized in accordance with the International Standards Organization (ISO) layered network model. The model describes seven levels and is officially known as the ISO Reference Model of Open Systems Interconnection, or simply the ISO Model. The model and many other interesting topics are discussed in Computer Networks by Andrew S. Tanenbaum.

The ISO model provides for layered processes, each supplying a set of services to a higher level process. The Pac-Comm TNC currently implements the first two layers, the Physical layer and the Data Link layer.

Physical Layer

The duty of the Physical Layer, layer one, is to provide for the transmission and reception of data at the bit level. It is concerned only with how each bit is physically transmitted, i.e., voltages on a hardwire line or modem tones on phone or RF links.

The physical layer of the Pac-Comm TNC is described in Chapter 7, Hardware. It is compatible with the various TNCs currently available to radio amateurs. The actual modem interface is compatible with the Bell 202 standard which is similar to the CCITT V.23 standard. Any other hardware device which is compatible with the Bell 202 standard should be compatible with the Pac-Comm TNC, at least at level one of the ISO reference model.

Data Link Layer

The duty of the Data Link layer is to supply an error-free stream of data to higher levels. Since level one simply passes any bits received to level two and is unaware of the content or overlying structure of the data, transmission errors are not detectable at level one. Level two carries the responsibility of detecting and rejecting bad data, retransmitting rejected data, and detecting the reception of duplicate data.

Level two accomplishes this task by partitioning data to be transferred by level one into individual frames, each with its own error detection field and frame identification fields. The Pac-Comm TNC supports two versions of a level-two layer, AX.25 version 1.0 and AX.25 version 2.0. Each of these protocols is based on HDLC, the High-Level Data Link Control protocol defined by the ISO.

HDLC Frames

Exact knowledge of the format of HDLC frames has been made largely unnecessary by the advent of LSI and VLSI communications chips which interface directly with the level one hardware. The level two software need only supply data to fill in various fields and the chip takes care of the rest. For completeness however, an HDLC frame looks like this:

FLAG	ADDRESS	CONTROL	PID & DATA	FCS	FLAG

FLAG — A unique bit sequence (01111110) used to detect frame boundaries. A technique called "bit stuffing" is used to keep all other parts of the frame from looking like a flag.

ADDRESS A field normally specifying the destination address. AX.25 uses a minimum of 14 bytes and a maximum of 70 bytes containing the actual call signs of the source, destination, and optionally up to eight digipeaters.

CONTROL A byte which identifies the frame type. In the AX.25 protocol, the control field may include frame numbers in one or two 3-bit fields.

PID A Protocol Identification byte appears as the first byte of the HDLC DATA field in AX.25 Level Two information frames, and identifies which Level 3 protocol is implemented, if any. In the case where no Level 3 protocol is implemented, PID = $F0.

DATA This field contains the actual information to be transferred. This field need not be present. Most frames used only for link control do not have data fields.

FCS Frame Check Sequence, a 16-bit error detection field.

The communications chip recognizes the opening and closing flags and passes the address, control, and data (including PID) fields to the software. The FCS field is a Frame Check Sequence computed by the transmitting chip and sent with the frame. The receiving chip recomputes the FCS based on the data received and rejects any frames in which the received FCS does not match the computed FCS. There is virtually no chance of an undetected bad frame using this method. This satisfies the level two task of bad data detection.

The communications chip used in the Pac-Comm TNC-220 is a Synchronous Communications Controller (SCC) (8530). The transmitted data is encoded in NRZI form, which encodes a "0" data bit as a transition in the encoded bit stream and a "1" data bit as no transition. This, in combination with the "bit-stuffing" which ensures that no more than five "1"s occur in a row except when FLAG bytes are being transmitted, guarantees that a logic level transition occurs at least once every 5 bit times. These frequent transitions allow the receiver to synchronize its clock with the transmitter. Other chips which are compatible with the SCC are the SIO + "state machine" (used on the TNC-200 and other TNC-2 type TNCs), the Western Digital 1933/1935 (used on the TAPR TNC 1, AEA PKT-1, Heathkit HD-4040, etc.), the Intel 8273 (used on the VADCG and Ashby TNCs).

While the HDLC format supplied by the communications chips is used by the AX.25 protocol, there are several other Layer Two concerns. These are duplicate frame detection, connection and disconnection of the level two layers on different TNCs, and buffer overrun avoidance. The AX.25 protocol solves these problems as described below.

AX.25 Level Two

AX.25 is based on the Balanced Link Access Procedure (LAPB) of the CCITT X.25 standard. LAPB in turn conforms to the HDLC standard. Two extensions are made to LAPB in AX.25. These are the extended address field, and the unnumbered information (UI) frame. In LAPB, addresses are limited to eight bits, while AX.25 uses from 112 to 560 bits, containing the originator's call sign, the destination call sign and an optional list of one to eight digipeater (simplex digital repeater) call signs.

The UI frame is used to send information bypassing the normal flow control and acknowledgment protocol. The UI frame is not acknowledged but can be transmitted at layer two without fear of disturbing higher layers. It is used for beacon frames, for automatic identification packets, and for sending information frames when the TNC is not connected to another TNC, e.g., CQ and QST activities.

The exact specifications for AX.25 are supplied in the ARRL publication Amateur Packet-Radio Link-Layer Protocol, AX.25 Version 2.0. The Pac-Comm implementation adheres to this standard for AX.25 version 2.0. The implementation of version 1.0 is almost identical to the TAPR TNC 1 version of AX.25 protocol in software releases 3.x. This provides compatibility with the majority of amateur packet radio stations.

The following table lists the frame types used by AX.25 and describes their purpose.

This material is provided to give a general understanding of the protocol, and is not intended to replace the published specification. The byte fields are given as they appear in memory after data is received, i.e., the high order bit is at the left and the low order bit is at the right. This is also the format of the display provided by the **TRACE** command. Some texts, including the AX.25 protocol specification, list the bits in the order in which they are transmitted, which is low order bit first.

The table also lists the format of the display of the interpertation of the command byte in each monitored frame when MCOM is turned ON. The basic structure of the MCOM information is-

< cmd_type [(C|R) (P|F)] [Sn] [Rn] >

The "<" and ">" characters are always present, and serve to delimit the new MCOM information. Cmd_type may take any of the values in the right column of the table.

C|R, P|F — The C(ommand), R(esponse), P(oll) and F(inal) frame types are only used in AX.25 level 2 version 2.0 mode. These types are not displayed when packets not using version 2.0 are monitored.

Sn — Sn is displayed for sequenced information (I) frames. 'n' is the frame's sequence number and is an integer 0..7.

Rn — Rn is present in both I frames and RR-RNR-REJ frames. The 'n' value monitored shows the sequence number that the sending station expects it's peer will use for the next new sequenced information frame.

The control bytes are presented in hex with "x" used to indicate four bits which depend on the acknowledge functions the packet is performing. Usually "x" is a frame number. Frame numbers fit into three bits and are used to ensure that frames are received in order and that no frames are missed. Since only three bits are available, the frame number is counted modulo 8. This is why the **MAXFRAME** parameter has a ceiling of 7: no more than seven frames can be "in flight" (transmitted but unacknowledged) at one time. A short description of the use of the frames is given after the table.

Table 9-1. AX.25 Control Codes and MCOM Display Codes

Code	Abbrv	Frame Type
x1	RR	Receive Ready
x5	RNR	Receive Not Ready
x9	REJ	Reject
03	UI	UnnumberedInformation
0F	DM	Disconnected mode
2F	SABM	Connect request
43	DISC	Disconnect request
63	UA	Unnumbered Acknowledge
87	FRMR	Frame reject
even	I	Any frame ending in an evennumber(including $A, $C, and $E) is an information frame.

Code	Definition
C	- SABM connect
D	- DISC disconnect
UA	- UA unnumbered acknowledge
DM	- DM disconnected mode
FRMR	- FRMR Frame reject
UI	- Unnumbered information, Beacons, ID frames, etc.
RR	- explicit acknowledgement, has R field
RNR	- receiver not ready, has R field
REJ	- reject, has R field
I	- Sequenced information packets, has both S and R fields.

I — This and UI frames are the only frame types containing user data. The control byte contains this frame's number and the number of the next frame expected to be received from the other end of the link.

RR — Usually used to acknowledge receipt of an I frame. The RR function can also be performed by sending an I frame with an updated "expected next frame number" field.

RNR — Used when the buffer space on the receiving side is full.

REJ	Used to request retransmission of frames starting from "x". Missed frames are detected by receiving a frame number different from that expected.
DM	Sent in response to any frame received other than a connect request (SABM) when the TNC is disconnected. Sent in response to an SABM when ever the TNC is on the air but can't connect to the requesting user, e.g., if the TNC is already connected to someone else or if **CONOK** is **OFF**.
SABM	Set Asynchronous Balanced Mode - initiates a connect.
DISC	Initiates a disconnect.
UA	Sent to acknowledge receipt of an SABM or DISC.
FRMR	Sent when an abnormal condition occurs, i.e., the control byte received is undefined or not proper protocol at the time received.
UI	An I frame without a frame number. It is not acknowledged.

Channel Use and Timing Functions

The following discussions mention timing parameters which are set by various commands. These timing functions are also discussed in Chapter 5.

An important part of any packet radio protocol is the means by which many stations make efficient use of an RF channel, achieving maximum throughput with minimum interference. The basis for this time domain multiplexing is Carrier-Sensed Multiple Access (CSMA) with collision detection and collision avoidance.

CSMA means simply that no station will transmit if the frequency is in use. The TNC continually monitors for the presence of packet frame flags on frequency and transmits only if there is no activity present. In order to make detection of a busy channel more reliable, the TNC sends an audio signal (continuous flags) any time the transmitter is keyed and a packet is not being sent, as during the transmitter keyup delay (**TXDELAY**), or while a slow audio repeater is being keyed (**AXDELAY**).

By itself, CSMA is not enough to insure a minimum, or even low, interference rate, due to the likelihood of simultaneous keyup by two or more stations. This is where collision detection and collision avoidance come in. The TNC detects a collision by the absence of an ACK from the station it is sending to. The receiving station does not acknowledge the frame that suffered the collision, since either the FCS was incorrect or the packet was not heard. There are other possible reasons for non-receipt of the packet, but the TNC's response is based on the assumption of a collision.

After transmitting a packet, the TNC waits a "reasonable" length of time (**FRACK**) for an acknowledgment. "Reasonable" is determined by the link activity, frame length, whether the packet is being digipeated, and other time-related factors. If no ACK is received, the packet must be re-sent. If the unACKed frame was lost due to a collision, the assumption is that there is at least one other packet station out there that also lost a frame and will probably have exactly the same criterion for deciding when to retry the transmission as this station is using.

In order to avoid a second collision, the collision avoidance protocol calls for the stations retrying transmissions to wait a random time interval after hearing the frequency become clear before they key their transmitters. There must be enough different random wait times to provide a reasonable chance of two or more stations selecting different values. The difference between adjacent time values must be similar to the keyup time delay of typical stations on the frequency. This is the time lapse after a station keys its transmitter before other stations detect its presence on the channel, and is a function of the keying circuitry of the transmitter and the signal detection circuitry of the receiver. We have chosen the random time to be a multiple (0-15) of the transmitting station's keyup delay (**TXDELAY**). This is reasonable if one's own keyup delay is similar to that of other stations on the channel.

One other factor must be taken into consideration in optimizing data throughput. The currently implemented link protocols provide for relaying (digipeating) of packets. The acknowledgment procedure for such packets is that the relay station simply repeats packets without acknowledgment to the sending station. The receiving station sends its ACK back through the same digipeaters to the originating station. Since the digipeated packets are not acknowledged to the digipeater, an unsuccessful transmission must be retried from scratch by the originating station. In order to help alleviate the congestion of the frequency that tends to result when digipeated packets suffer collisions, the digipeater is given first shot at the frequency every time it becomes clear. Other stations, instead of transmitting as soon as they hear the channel clear, must wait a short time (**DWAIT**). This restriction applies to all stations except the digipeater, which is permitted to transmit relayed packets immediately. This prevents digipeated packets from suffering collisions except on transmission by the originating station.

A special time delay (**RESPTIME**) is used as the minimum wait time prior to transmitting acknowledgment frames, to prevent TNCs accepting data at high speed from the asynchronous port from colliding with acknowledgment frames when fewer than **MAXFRAME** packets are outstanding. The receiving TNC will wait long enough before sending the ACK so that it will hear the data packet which would have caused the collision, thus avoiding a fairly frequent source of delay in versions of AX.25 prior to 2.0.

Channel Flow Control

Flow control of data through the link is determined by the rate at which data is being supplied to a sending TNC and accepted from a receiving TNC.

A TNC receiving data from the link will send an RNR when the next I frame successfully received will not fit into the buffer for output to the serial port.

Whenever a TNC transmitting data received from the serial port over the link runs out of temporary buffer space, the serial port will be halted by an XOFF character or CTS signal. In the TNC-220 implementation this happens whenever there are 7 packets built and less than 210 characters left in the buffer for input from the serial port.

When the TNC receiving data from the link clears out its buffers, it sends an RR to the transmitting TNC. In order to guard against the possibility of the RR being lost and the link becoming permanently locked, the transmitting TNC will periodically re-transmit the packet that provoked the RNR. The receiving TNC will continue to respond with RNR until it can accept the packet.

PAC-COMM TNC-220 REFERENCE MANUAL, APPENDIX A, COMMODORE 64 INTERFACING

There are several levels of interfacing possible between the TNC-220 and the Commodore-64 and similar TTL computers such as the VIC-20 and C-128. If you plan to use your C-64 as a simple terminal for only keyboard to keyboard QSOs, then the 3 wire cable will be sufficient. The three wire interface is also adequate for applications requiring flow control if 'software' flow control is implemented in your terminal program. The five wire interface will allow hardware handshaking. Also, 'CONNECTED' status information is available on J3 pin 8 for a Bulletin Board system.

Choose the level of interface that meets your needs and follow the directions below.

1. Place the TNC-220 jumper JP-T shorting block in the 'TTL' position.

2. Obtain the following cable components:
   - DB-25P solder cup connector on TNC-220 end.
   - 12/24 pin 'Commodore' card edge connector at C-64 end.
   - Shielded wire: 4 or 5 conductor for handshaking cable.
     2 or 3 conductor for non-handshaking cable.

3. Wire a cable to the specifications for the level of interface desired.

FUNCTION	DB-25	12/24	CABLE TYPE
FRAME GROUND	1	N+12	BOTH 3 AND 5 CONDUCTOR (USE CABLE SHEILD)
TRANSMIT DATA	14	M	BOTH 3 AND 5 CONDUCTOR
RECEIVE DATA	16	B+C	BOTH 3 AND 5 CONDUCTOR
REQUEST TO SEND	19	D	5 CONDUCTOR ONLY
CLEAR TO SEND	13	K	5 CONDUCTOR ONLY
SIGNAL GROUND	7	A+1	BOTH 3 AND 5 CONDUCTOR (MAY BE TIED TO PIN 1)
DATA CARRIER DET	12	H	OPTIONAL (CONNECTED STATUS)

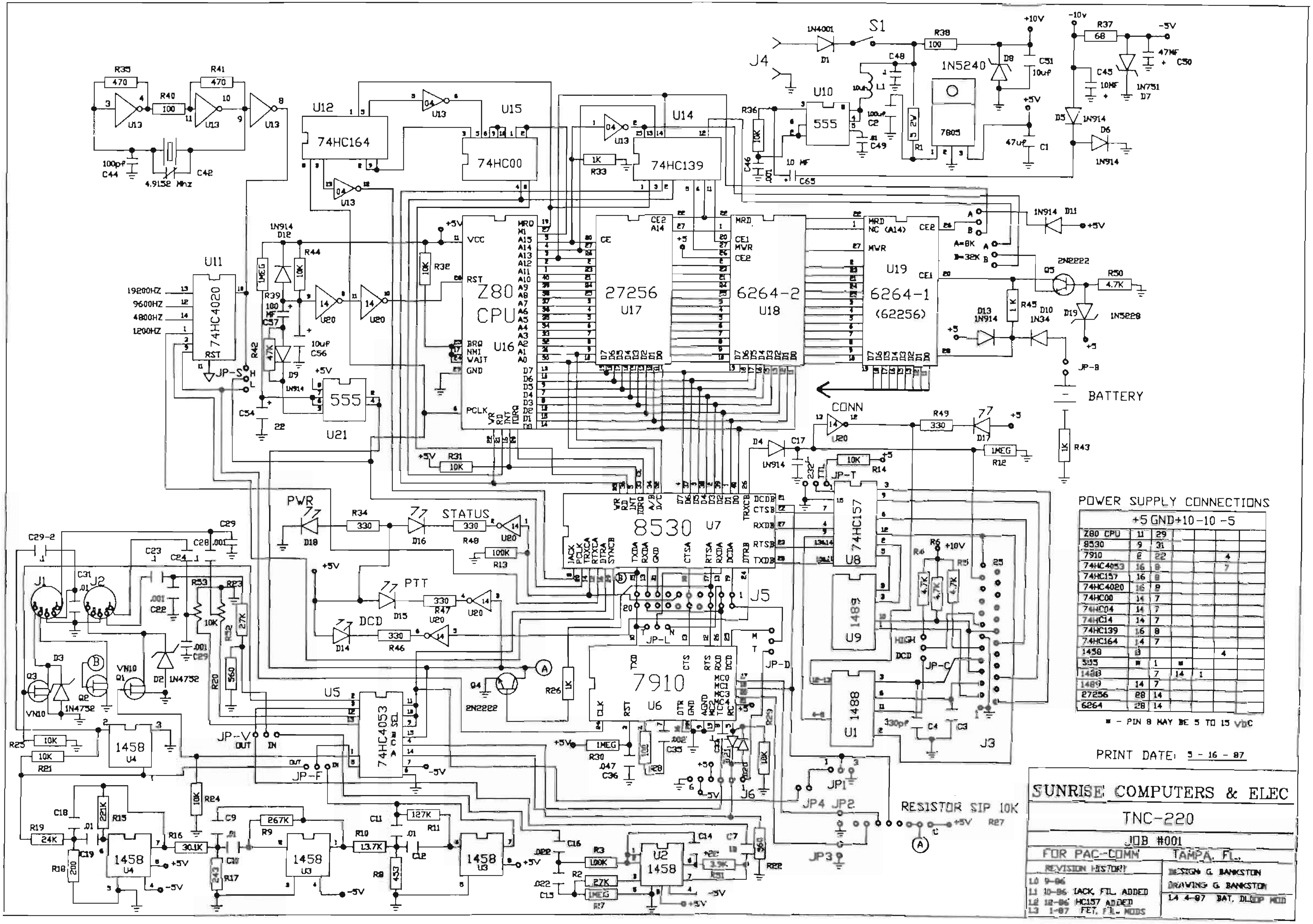
NOTE: You must jumper TNC-220 pins 13 and 19 together in a non-handshaking cable. If you try to use a handshaking cable and your C-64 software does not support handshaking, the TNC-220 will not communicate with your computer.

4. Install the cable and load a terminal program in the computer (A simple BASIC program is listed below for your use in checking out the TNC-220). Follow the directions with your terminal program and refer to Chapter 2 in the Reference Manual.

```
10 REM SIMPLE C64 TERMINAL PROGRAM (OPERATOR MUST USE CTL-S AND CTL-Q FOR FLOW CONTROL)
15 REM F1 SENDS A CONTROL C TO TNC
20 POKE53280,1:POKE53281,1:POKE53272,23
25 REM Use this line 20 for VIC-20: 20 POKE36879,25
30 OPEN2,2,3,CHR$(40)+CHR$(96)
40 POKE36869,242
45 IF A$=CHR$(133) THEN A$="CONTROL C"
50 PRINT CHR$(144)
60 C$=CHR$(147)
70 K$=CHR$(20)
80 J$=CHR$(167)
90 CR$=CHR$(13)
100 PRINT C$;"WAIT..."
110 GOSUB300
120 PRINT C$;"POWER-UP YOUR TNC-220 NOW"
130 PRINTJ$;
140 GET#2,A$
150 IF A$="" THEN 190
160 A=I%(ASC(A$))
170 PRINT K$+CHR$(A)+J$;
180 GOTO 140
190 GET A$
200 IF A$="" THEN 140
210 PRINT#2,CHR$(O%(ASC(A$)));
220 GOTO 140
300 DIMI%(255),O%(255)
310 FORZ=0TO64:O%(Z)=Z:NEXT:O%(13)=13:O%(20)=8:O%(160)=32
320 FORZ=65TO90:Y=Z+32:O%(Z)=Y:NEXT:FORZ=91TO95:O%(Z)=Z:NEXT
330 FORZ=193TO218:Y=Z-128:O%(Z)=Y:NEXT
340 O%(133)=03:O%(134)=19:O%(135)=17:O%(136)=16
350 FORZ=0TO255:Y=O%(Z):IFY<>0THENI%(Y)=Z
360 NEXTZ
370 RETURN
```

NOTE: This program is offered for the initial convenience of the new TNC-220 owner to enable checkout and beginning operation of the TNC. The program is believed to operate correctly on the C-64 and C-128 in 64 mode, and has been used on the VIC-20 with a changed line 20.

The statement that the TNC-220 is compatible with the C-64, Vic-20, and C-128 refers to the inclusion of a TTL port on the TNC and not to this simple program. Terminal software is not included in the price of the TNC and is the responsibility of the owner.



POWER SUPPLY CONNECTIONS

	+5	GND	+10	-10	-5	
Z80 CPU	11	29				
8530	9	31				
7910	2	22			4	
74HC4053	16	8			7	
74HC157	16	8				
74HC4020	16	8				
74HC00	14	7				
74HC04	14	7				
74HC14	14	7				
74HC139	16	8				
74HC164	14	7				
1458	8				4	
555	■	1	■			
1488		7	14	1		
1489	14	7				
27256	28	14				
6264	28	14				

■ - PIN 8 MAY BE 5 TO 15 VDC

PRINT DATE: 5 - 16 - 87

SUNRISE COMPUTERS & ELEC

TNC-220

JOB #001

FOR PAC-COMM TAMPA, FL.

REVISION HISTORY

1.0 9-86
1.1 10-86 IACK FIL ADDED
1.2 12-86 HC157 ADDED
1.3 1-87 FET, FIL. MODS

DESIGN G. BANKSTON
DRAWING G. BANKSTON
1.4 4-87 BAT, DLOOP MOD

APPENDIX C

TNC220 PARTS LIST

Board Revision 1.4, August 27, 1987

RESISTORS

Qty	Description	Location
1	5 ohm, 2 watt	R1
1	68 ohm, 1/4 w, 5%	R37
3	100 ohm, 1/4 w, 5%	R28, R38, R40
1	200 ohm, 1/4 w, 1%	R18
1	243 ohm, 1/4 w, 1%	R17
5	330 ohm, 1/4 w, 5%	R34, R46-49
1	453 ohm, 1/4 w, 1%	R8
2	470 ohm, 1/4 w, 5%	R35, R41
2	560 ohm, 1/4 w, 5%	R20, R22
4	1K ohm, 1/4 w, 5%	R26, R33, R43, R45
1	3.9K ohm, 1/4 w, 5%	R51
4	4.7K ohm, 1/4 w, 5%	R4-R6, R50
9	10K ohm, 1/4 w, 5%	R14, R21, R24, R25, R29, R31, R32, R36, R44
1	13.7K ohm, 1/4 w, 1%	R10
1	24K ohm, 1/4 w, 5%	R19
2	27K ohm, 1/4 w, 5%	R2, R52
1	30.1K ohm, 1/4 w, 1%	R16
1	47K ohm, 1/4 w, 5%	R42
2	100K ohm, 1/4 w, 5%	R3, R13
1	127K ohm, 1/4 w, 1%	R11
1	221K ohm, 1/4 w, 1%	R15
1	267K ohm, 1/4 w, 1%	R9
4	1 Megohm, 1/4 w, 5%	R7, R12, R30, R39
2	10K ohm trimpot	R23, R53
1	10K 8 pin resistor	R27

CAPACITORS

Qty	Description	Location
1	100 pF	C44
2	330 pF	C3, C4
4	.001 uF	C22, C25, C29, C46
1	.002 uF	C35
10	.01 uF	C9, C10, C11, C12, C18, C19, C26, C30, C31, C49
2	.022 uF	C15, C16
1	.047 uF	C36
31	.1 uF	C5, C6, C8, C13, C17, C20, C21, C23, C24, C28, C32, C33, C37-41, C43, C47, C48, C52, C53, C55, C58, C59, C60, C61, C62, C64, C66, C67
5	10 uF electrolytic	C7, C45, C51, C56, C65
2	22 uF	C14, C54
2	47 uF	C1, C50
2	100 uF (one axial)	C2, C57
1	50 pF trimmer	C42

NOTE: There is no C27, C34, or C63

Qty	Description
1	Extruded aluminum cabinet, med gray
1	set (2) plastic cabinet bezels
1	set (2) front and rear end panels
1	TNC-220 rev 1.4 circuit card
1	Reference Manual
1	ERRATA/Assembly Notes
1	Software Notes
1	Command Reference Card

INTEGRATED CIRCUITS

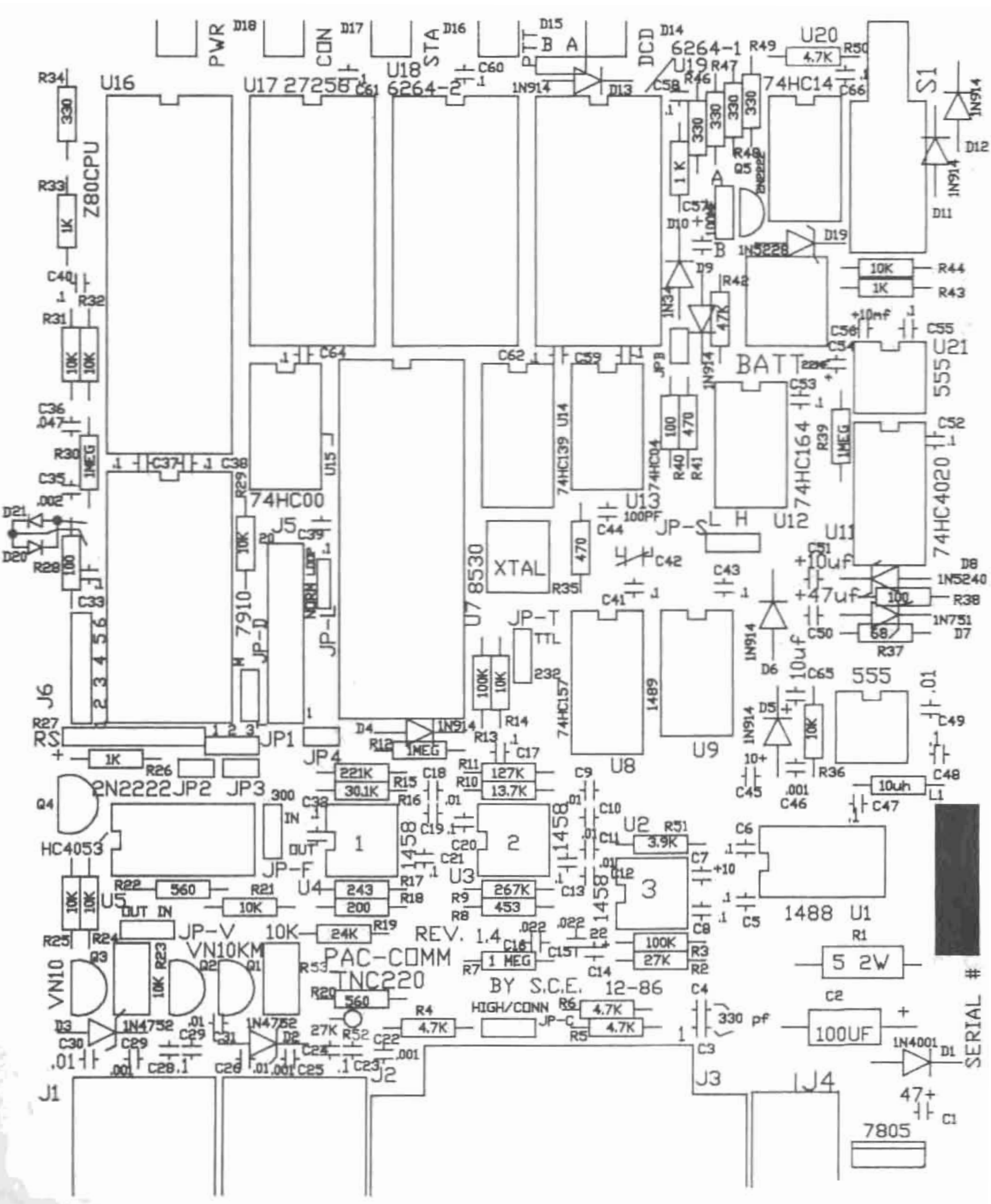
Qty	Description	Location
2	LM555 or LM1455 timer	U10, U21
3	LM1458 op amp	U2, U3, U4
1	1488 RS-232 driver	U1
1	1489 RS-232 receiver	U9
1	74HC00 quad NAND	U15
1	74HC04 hex inverter	U13
1	74HC14 hex inverter	U20
1	74HC139 dual 2-4 mux	U14
1	74HC157 quad 2-in mux	U18
1	74HC164 shift register	U12
1	74HC4020 divider	U11
1	74HC4053/4053 analog switch	U5
1	8530 SCC	U7
1	7910 modem	U6
1	27256 EPROM	U17
2	5564/6264 8K RAM	U18, U19
1	20256/43256/62256 32K RAM	U19 (optional)
1	Z80 CPU	U16

SOCKETS

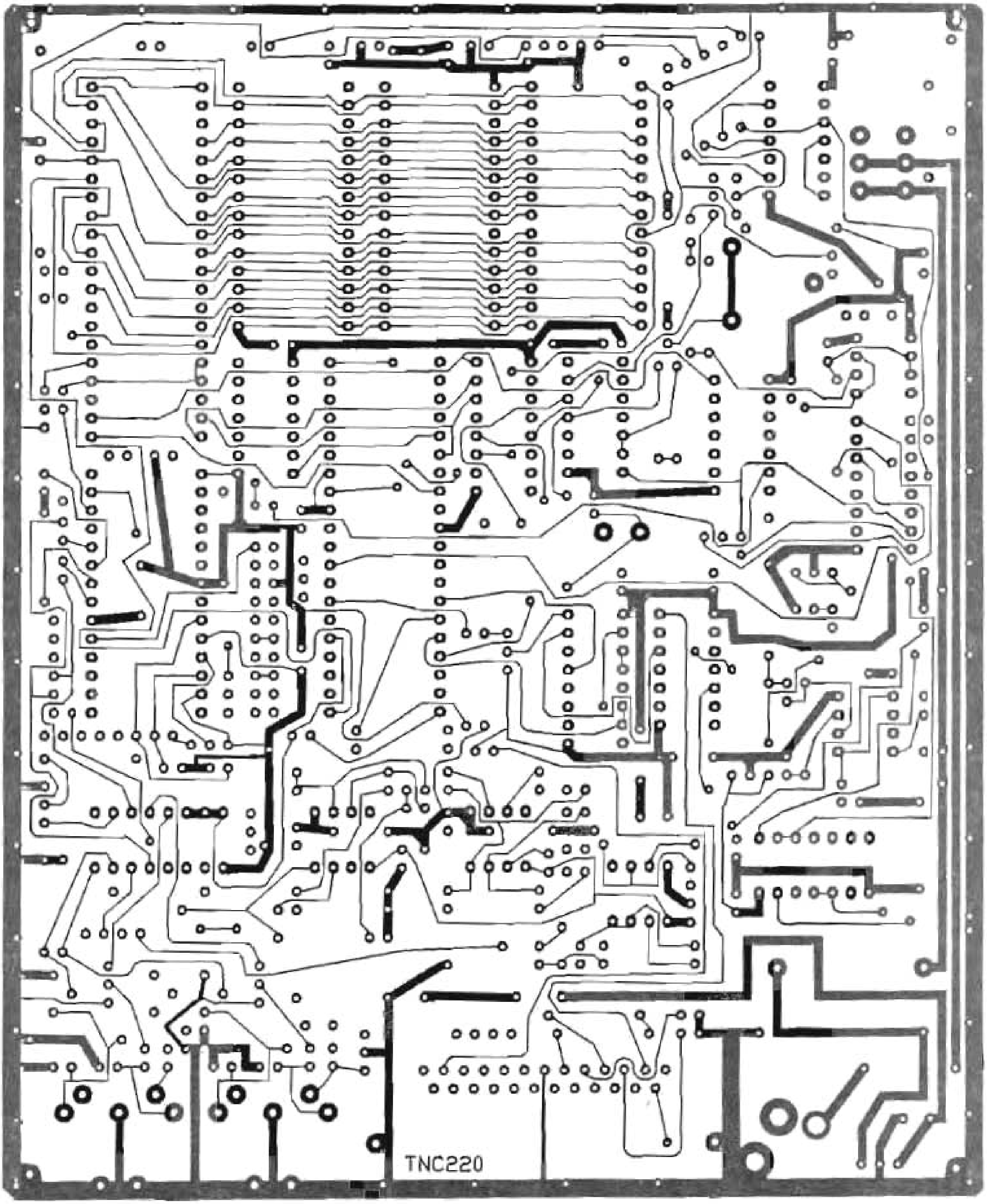
Qty	Description
5	8 pin
6	14 pin
4	16 pin
4	28 pin
2	40 pin

MISCELLANEOUS

Qty	Description	Location
1	10uH choke	L1
5	LED (2 green, 2 red, 1 yellow)	D14-D18
1	1N751 5v Zener	D7
9	1N914 signal diode	D4-6, 9, 11-13, /20-21
1	1N34 germanium diode	D10
2	1N4752 35 v Zener	D2, D3
1	1N5228 3.9v Zener	D19
1	1N5240 10v Zener	D8
1	1N4001/4004 rectifier	D1
2	2N2222 NPN	Q4, Q5
3	VN10KM or SD1106 FET	Q1-Q3
1	7805 5v regulator	
1	4.9152 MHz XTAL	
1	3.2 V Lithium battery	
2	4-40 x 3/8 screws and nuts	
1	Push on/off switch	
1	Clip-on heat sink (paper clasp)	
4	Stick-on rubber feet	
4	#4 x 3/4 self tapping screws	
1	#6 x 3/8 self tapping screw	
4	2 pin male headers	
10	3 pin male headers	
2	5 pin female DIN sockets, PC mount	
1	DB-25 socket, right angle PC mount	
1	2.1 mm coaxial power socket, PC mount	
13	push-on jumpers (shorting blocks)	
2	5 pin male DIN cable connectors	
1	2.1mm coaxial power plug	
1	6 pin walled connector	



Appendix D-I TNC-220 Revision 1.4 Board Layout



TNC-220 Component Side Traces, Board Revision 1.4
Appendix D-2

TNC-200 and TNC-220 RADIO CABLE PREPARATION

The TNC-200 and TNC-220 radio port(s) are designed to work with almost all common (and many not so common) transceivers or transmitter-receivers.

Since there is such a wide variety of radio connectors in use in Amateur radio, the mating connector for your radios, as well as the wire to connect to them, are not included in the TNC-200 and TNC-220 kits. However, we have included mating connectors for the radio port(s) on the TNC, along with cable boot(s).

To construct the radio interface cable, you will need to obtain a mating connector(s) for your radio's microphone and headphone or external speaker jack(s). We recommend you keep the length of the radio interface cable to under about four feet (1.2 m). Use shielded wire for all audio connections.

CAUTION: In the following steps, use only the minimun heat necessary when soldering to the pins of the DIN connector. Excessive heat will soften the body of the DIN plug and allow the pins to become misaligned.

( ) Locate the 5-pin male DIN connector(s).

( ) Prepare a length of shielded wire by separating the braided shield from the insulated center conductor, then strip about 1/4" of insulation from the center conductor.

( ) Solder the center conductor to the audio output terminal of your radio headphone/ speaker connector.

( ) Solder the shield to the audio common (or negative power common, NOT audio common if audio common is at a positive potential with respect to your radio's power "ground").

( ) Slide the backshell of your headphone/speaker connector over the shielded wire and fasten to the connector.

( ) Pass the shielded wire through the insulating boot of the DIN connector and carefully solder the center conductor to pin 4 of the DIN connector.

( ) Prepare another length of shielded wire.

( ) Solder the center conductor to the microphone audio input contact of your radio microphone connector.

( ) Solder the shield to the microphone audio common (this common MUST be at the same dc potential as the headphone/speaker audio common as well as dc power common).

( ) Slide the backshell of your microphone connector over the shielded wire.

( ) Pass the shielded wire through the insulating boot of the DIN connector and carefully solder the center conductor to pin 1 of the DIN connector.

( ) Prepare an unshielded wire the same length as the microphone audio wire above.

( ) Solder one end of this wire to the PTT (transmitter key) contact of your radio microphone connector.

( ) Pass this wire through the microphone connector backshell and secure the backshell to the microphone connector.

( ) Pass the shielded wire through the insulating boot of the DIN connector and carefully solder the center conductor to pin 3 of the DIN connector.

( ) Carefully solder both shields to pin 2 of the DIN connector.

( ) Inspect the DIN connector for any stray wires or other potential causes of shorts within the connector.

( ) Slide the insulating boot over the DIN connector to secure the connector assembly.

( ) With your radio and TNC-200 or TNC-220 switched off, attach the microphone and headphone/speaker connectors to your radio.

( ) Insert the DIN plug in the proper DIN receptacle on your TNC-200 or TNC-220.

PacComm Manual Errata and Software Release Notes, April 22, 1991

Updates and corrections to the PacComm Operating Manual, various Technical Reference and Hardware Manuals. The current version of each manual is listed. This sheet applies to all PacComm products. Look for the section which applies to the product which you received.

Firmware release (Version 1.1.6D4, PMS 3.0) is now shipping on all products. This release contains fixes for the following customer reported bugs: Failure of the PMS to send a SUBJ: prompt; Budlist and LCalls were inoperative; KISS mode failure.

ALL PRODUCTS: Power connectors are center pin positive.
Current version of operating instructions for all PacComm amateur products except the PC-100 series:
Operating Manual for PacComm Packet Controllers, 4th ed., 2nd printing, April 1991.

TINY-2 and MICROPOWER-2:
The RFDCD input on these products is different than on the TNC-2 and clones. Contact PacComm for an Application Note to correct this problem.

Current Firmware:	TINY-2 without PMS:	P1.1.6D4 $26
	TINY-2 with PMS:	P1.1.6D4 PMS 3.0 $FC
	European TINY-2 with PMS:	E1.1.6D4 PMS 3.0 $56
	MICROPOWER-2/PMS:	P1.1.6D4 PMS 3.0 $EE
	MICROPOWER-2/PMS/PRT:	P1.1.6D4 PRT PMS 3.0 $9A
	European MICROPOWER-2/PMS:	E1.1.6D4 PMS 3.0 $8A
	European MICROPOWER-2/PMS/PRT:	E1.1.6D4 PRT PMS 3.0 $2C

Technical Reference Manual for the TINY-2 and MICROPOWER-2, 2nd Ed., 2nd Printing, Feb. 1991.

PC-320: Current Firmware: P1.1.6D4 PMS 3.0 $B2
European PC-320 E1.1.6D4 PMS 3.0 $2F
Technical Reference Manual for the PC-320, First Ed., First Printing, August 1989:

Page 6, Radio Jack (S2) should be labeled "Pin definitions viewed from PC-320 into end of cable connector."

Page 8, Default Terminal Baud Rate is **4800** (instead of 19,200)

Page 10: JPI1: LOWER selects IRQ5 (instead of 7)
JPI2: LOWER selects IRQ7 (instead of 2)
There is a bug in the KISS module which prevents proper HF (Port 1) operation with KISS. Call PacComm for information on a temporary fix until the bug is corrected.

TNC-320: Current Firmware: P1.1.6D4 PMS 3.0 $65
European TNC-320 E1.1.6D4 PMS 3.0 $6A
There is a bug in the KISS module which prevents proper HF (Port 1) operation with KISS. Call PacComm for information on a temporary fix until the bug is corrected.

HandiPacket: Current Firmware: P1.1.6D4 PMS 3.0 $A2
European HandiPacket E1.1.6D4 PMS 3.0 $36
Technical Reference Manual for the HandiPacket, 1st Revised Ed., Second Printing, February 1991:

Radio cable color coding:	Shell - Shield	Pin 5 - Green
	Pin 1 - Brown	Pin 6 - Blue
	Pin 2 - Red	Pin 7 - Violet
	Pin 3 - Orange	Pin 8 - Black
	Pin 4 - Yellow	

TNC-200: Current Firmware, 32k RAM: P1.1.6D4 $30; P1.1.6D4 PMS 3.0 $2F (with PMS)
European TNC-200, 32k RAM: E1.1.6D4 $DF; E1.1.6D4 PMS 3.0 $41 (with PMS)

TNC-220: Current Firmware, 32k RAM: P1.1.6D4 $0C; P1.1.6D4 PMS 3.0 $1D (with PMS)
European TNC-220, 32k RAM: E1.1.6D4 $9F; E1.1.6D4 PMS 3.0 $8F (with PMS)
Note: The SOFTDCD command is defaulted OFF, but must be set to ON for proper operation if there is no tuning indicator installed.

MC-NB96: Technical Reference Manual, Second Edition.

The MC-NB96 is shipped without the mounting connector attached to the circuit board. Solder the connector at location S1A for the TNC-200 (TNC-2, TNC-2A, PK-80), MFJ 1270/1270B/1274/1278, and PC-320. Solder the connector at location S1 for the TINY-2, MICROPOWER-2, and TNC-320. Insure that the pin 1 of the connector contacts pin 1 of the disconnect header. Pins 21-26 of the connector hang over the edge of the 20 pin disconnect header. The 26 pin connector may be shortened to 20 pin size with a knife or saw if necessary for clearance.

PacComm packet controllers have pin 1 of the modem disconnect header located as follows: TNC-200, TINY-2, MICROPOWER-2: Pin nearest to board edge;TNC-220: pin closest to rear of circuit board (marked); TNC-320:Pin closest to J5 marking; PC- 320: pin closest to lower edge of circuit board.
Cut traces 1 &2, 13 &14, 17 &18, 19 &20 on the underside of the modem disconnect header of the packet controller.

EM-NB96: The MC-NB96 manual was provided with the EM-NB96. A new manual is being written. The EM-NB96 is currently undergoing revision. The new revision will support the 32x clock of AEA products and the 'daisy chaining' of other accessory modems to the attached TNC. If you need to connect the current (revision 1.1) EM-NB96 to a PK-232 or other AEA product, please contact PacComm for assistance.

Jumper 4 on the EM-NB96 is a watchdog timer control. If the jumper is installed, the watchdog timer is disabled.

PC-100 Series (PC-105, PC-110, PC-120):
Node Manager Disks 1-2-3: April 20, 1990, Manual First Ed., April 1990

PSK-1: Current Firmware: Version 2.01, 4/21/90 $6717
Technical Reference Manual, First Edition, March 1990: (Revised AEA and Kantronics Appendices, April 1991)

Page 8, Serial Port Connections: Note that two serial ports on the computer or an A/B serial switch are needed to make use of the PSK-1 telemetry/control port. One computer serial port is needed for connection to the packet controller. All incoming and outgoing packet information go through the packet controller, so the serial connection to the packet controller must always be intact. Control and telemetry information bypass the packet controller and go directly from J10 to the computer serial port.

Page 29, Jumper Definitions: JP-4 is better described as an output level test point. JP-5 gives increased output level when jumpered. JP-6 is located close to the two voltage regulators. Position 5&6 (for PK-232) is the pair of pins closest to the center of the circuit board. Position 1&2 are located closest to the edge of the board.

For increased PSK-1 output, Install JP-5, change R41 to a lower value (not less than 13k) and change C66 from 0.1uf to 0.2 or 0.5 uf. One final change is to lift the ground leg of R40 and insert a 5k resistor to ground.

Page 42-43, Appendix C, PacComm and MFJ Interfacing:
- Cut the following traces on the underside of the modem disconnect header on the packet controller: 1 &2, 17 &18. Factory installed units will also have pins 13/14 and 19/20 cut and then tied together on the cable connector..
- PacComm packet controllers have pin 1 of the modem disconnect header located as follows: TNC-200, TINY-2, MICROPOWER-2: Pin nearest to board edge;TNC-220: pin closest to rear of circuit board (marked); TNC-320:Pin closest to J5 marking; PC- 320: pin closest to lower edge of circuit board.
- Contact PacComm for assistance when interfacing to TNC-220 and TINY-2/MICROPOWER- 2 units of revision 1.4 and earlier.
- The TNC-320 and PC-320 have a 26 pin header. If connecting a 20 pin connector, gently bend pins 21-26 out of the way to allow the connector to fit. Be sure the bent pins are not shorted together. The 26 pin connector may be shortened to 20 pin size with a knife or saw if the extra long connector interferes with components on the TNC.

- Ignore references to Note 5 in the pin definition table.

Page 44-45, Appendix D, AEA Interfacing:
Ignore this Appendix and follow the instructions on the loose sheet provided with the manual.

Page 46-47, Appendix E, Kantronics Interfacing:
Ignore this Appendix and follow the instructions on the loose sheet provided with the manual.

Page 49-50, Appendix F, Radio Interfacing:
- The diagrams for Kenwood radio interfacing are mistakenly shown under the Yaesu section. The Yaesu diagrams near the top of page 50 should be on the lower part of page 49. The lower diagram on page 50 is for the Yaesu FT-726 only, not the FT-736.

PSK-1 cable color coding: (Confirm with Ohmeter.)

	8 pin DIN cable	5 pin DIN cables
Pin 1 -	Red	Yellow
Pin 2 -	Yellow	Red
Pin 3 -	Blue	Black
Pin 4 -	Orange	Orange
Pin 5 -	Green	Brown
Pin 6	Brown	
Pin 7	Violet	
Pin 8	Black	