

**OPERATING
INSTRUCTIONS**

**TINY-2
and
MICROPOWER-2
Packet Controllers**



PacComm

OPERATING MANUAL

for

PacComm Packet Controllers

**PacComm Packet Radio Systems, Inc.
4413 N. Hesperides Street
Tampa, FL 33614-7618**

**Business Office (813) 874-2980
Technical Support (813) 875-6417**

**Electronic Mail System (813) 874-3078
Telefax (813) 872-8696
TELEX 650-288-1526 MCI**

**Toll free order line (orders only, please)
(800) 223-3511**

**This entire manual is Copyright © 1991 by
PacComm Packet Radio Systems, Inc.
ALL RIGHTS RESERVED**

Reproduction or translation of any part of this work beyond that permitted by Sections 107 or 108 of the 1976 United States Copyright Act (or its legal successor) without the express permission of the copyright owner is unlawful. Requests for permission to copy or for further information should be addressed to PacComm Packet Radio Systems, Inc.

Fourth Edition, Fifth Printing - October 1991

PacComm Warranties

PacComm Packet Radio Systems, Inc. (PacComm) warrants amateur 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, PacComm 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, used in commercial service or which has been repaired outside PacComm facilities without permission. Consequential damages are not covered.

Repair Service

PacComm provides a rapid, low cost repair service for our products, whether in or out of warranty. Carefully pack the unit and send it to the PacComm 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.

WARNING: The packet controller 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. There are no lethal voltages present in the packet controller. Careful connection of the unit to power sources and radio and computer equipment will insure that no hazardous situation is created through the interconnection. Carefully ground all units to a solid earth ground. When operating the packet controller out of its cabinet, avoid touching the voltage regulator heat sink, which may become hot.

TABLE OF CONTENTS

INTRODUCTION	1
Computer and Radio Interfacing	1
Software Requirements	2
Radio Interfacing.....	2
Other Information.....	2
OPERATING INSTRUCTIONS	3
Verifying Serial Port Operation	3
Serial Port parameters	4
Beginning Packet Operation	6
First Steps	6
Your First Packet Connection.....	6
Starting the Connection.....	7
In Case of Difficulty	8
Digital Repeating (Digipeating).....	8
Unsuccessful Connections.....	10
Monitoring Channel Activity	11
Special Characters.....	12
Special Input Characters	13
Operating Modes.....	15
Command Mode.....	15
Entering Data-Transfer Modes	16
KISS Mode	17
Converse Mode.....	18
Transparent Mode	19
Flow Control	20
XON/XOFF Flow Control	20
Hardware Flow Control.....	21
Type-in Flow Control	21
Station Identification	22
Automatic Operations	22
Unattended Connections.....	22
Beaconing.....	23
Connect Alarm.....	24
Packet Formatting	24
Commands Affecting Protocol	24
Packet Transmit Timing	25
Packet Timing	25
Alternate Transmit Timing.....	26
Special Protocol Times	27

Monitor Functions.....	27
“Health” Features.....	28
Clock Operations.....	31
Time Stamping.....	31
Special Operations.....	31
Computer Operations	32
HF Operation.....	32
Multiple Connections.....	33
Personal Message System (PMS)	34
PMS Setup Commands	35
PMS Operator Commands	36
PMS User Commands.....	38
PACKET RADIO PROTOCOL.....	39
Physical Layer	39
Data Link Layer.....	39
HDLC Frames.....	40
AX.25 Level Two	41
Channel Use and Timing	43
Channel Flow Control.....	45
PACKET CONTROLLER COMMANDS	46
Parameters.....	46
Command Groups	48
Command List.....	50
MESSAGES.....	98
General Messages.....	98
Command Mode Error Messages.....	99
Link Status Messages	100
PMS Messages	102
PMS Response Messages.....	102
PMS Error Messages.....	104
INDEX.....	106

TECHNICAL REFERENCE MANUAL follows page 109

NOTE: Please be sure to read any Errata, manual changes or software notes that accompany your manual and packet controller. These documents will describe all changes and upgrades made to your unit.

PacComm Amateur Products International Distributors and Agents

Contact the Distributor or Agent in your area for dealer names and locations. Contact PacComm for areas not listed.

Distributor for Europe, Western Asia, and the Middle East:

Europdata Systems, NV SA Telephone: 32-3-235-3625
Dr. Van De Perrelei, 92 Telefax: 32-3-235-8031
2140 Borgerhout, Belgium

Sales Agent for Argentina:

Special Communications Telephone: 541-46-6588
Rodriguez Pena 434-5 "A" Telefax: 541-46-6395
1020 Capital Federal
Buenos Aires, Argentina

Sales Agent for Chile:

Federhit S.A. Telephone: 563-2-972-104
Quillota 853, Casilla 803 Telefax: 563-2-660-294
Vina Del Mar, Chile

Distributor for New Zealand:

Ryel Electronics, Ltd. Telephone: 64-3-351-5630
P.O. Box 29-167, Fendalton Telefax: 64-3-348-8294
Unit 2, 65 Manderville St., Christchurch 1, New Zealand

Sales Agent for Australia:

Blamac Pty Ltd Telephone: 61-64-523-112
26B Bombala St. Telefax: 61-64-524-317
Cooma, NSW 2630 Australia:

Sales Agent for South Africa:

Christopher Turner/ Telephone: 27-11-802-8576
P.O. Box 19, Wendywood Telefax: 27-11-804-4729
2144 Johannesburg, RSA

This Operating Manual is dedicated to Tom Ringate, K0ZXF. Tom has devoted hundreds of hours of his spare time supporting amateur packet radio. He is a local BBS Sysop, ROSE switch operator, and leader in local network planning.

PacComm thanks Tom for his efforts in developing and testing the PacComm Personal Message System. Through his skill and vision, many features and capabilities have been added which make packet radio more fun for all of us. His work to bring automated message handling to the terminal node makes possible significant reduction of channel congestion during prime packet operating hours.

U.S. Federal Communications Commission Notice to the Equipment Operator

The FCC places limits on the strength of non-intentional signals emitted by electronic accessories which attach to, or install in, personal computers. These computer peripherals are classified as "Class B Computing Devices." The following notice is required by law.

This equipment generates and uses radio frequency energy. If not installed and used properly, that is, in strict accordance with the manufacturer's instructions, it may cause interference to radio and television reception.

It has been type tested and found to comply with the limits for a Class B computing device in accordance with the specifications in Subpart J of Part 15 of FCC Rules, which are designed to provide reasonable protection against such interference in a residential installation. However, there is no guarantee that interference will not occur in a particular installation.

If this equipment does cause interference to radio or television reception, which can be determined by turning the equipment off and on, the user is encouraged to try to correct the interference by one or more of the following measures:

- *Reorient the receiving antenna*
- *Relocate the computer with respect to the receiver*
- *Move the computer away from the receiver*
- *Plug the computer into a different outlet so that computer and receiver are on different branch circuits.*

If necessary, you should consult the dealer or an experienced radio/television technician for additional suggestions. You may find the following booklet prepared by the Federal Communications Commission helpful: *How to Identify and Resolve Radio-TV Interference Problems*. This booklet is available from the US Government Printing Office, Washington, DC 20402. Stock No. 004-000-00345-4.

Excellent information about the elimination of interference is contained in *The ARRL Handbook For The Radio Amateur* and other specialized publications dealing with RFI/TVI from the American Radio Relay League, Newington, CT 06111.

The user of this equipment must follow the requirements in the **HARDWARE MANUAL** for selection or construction of cables used to interconnect the computer or terminal and the radio to this equipment to insure compliance with the notice above.

INTRODUCTION

The purpose of the PacComm packet controller is to act as an interface between an ordinary voice radio, such as a VHF or UHF FM transceiver, and a computer. The packet controller will establish error-free radio communications between your computer and another packet-radio equipped station. You will be able to have a "private channel" while sharing a frequency with other packet stations, operate remote computer "bulletin board" or "electronic mail" stations, send messages, — in short, to enjoy all the advantages of digital communication via your radio system.

This OPERATING MANUAL explains how the packet controller operates and contains a detailed breakdown of the various commands the packet controller will accept and messages it may report, as well as a description of packet radio protocol.

The COMMAND REFERENCE CARD contains a listing of packet controller commands in a convenient form for reference while you are operating. These two documents are written to apply to the entire line of PacComm amateur products, and may contain descriptions of commands not included in the firmware on every model.

There is a TECHNICAL REFERENCE MANUAL for each series of PacComm packet Controller. Refer to the TECHNICAL REFERENCE MANUAL for details on interfacing your packet controller to specific computers and radios. The TECHNICAL REFERENCE MANUAL also contains details about the specific features of the packet controller to which it applies.

The term "packet controller" was chosen to refer to the PacComm equipment throughout this manual. Other terms commonly used are packet assembler/disassembler, terminal node controller (TNC), packet modem, packet interface, etc. The term "computer" is used to refer to the computer or terminal which is connected to the packet controller.

Computer and Radio Interfacing

The packet controller communicates with your computer through a serial port using RS-232C signal levels. Some models also support TTL level signals to interface to such popular computers as the Commodore VIC-20, C-64, and C-128 without requiring the use of an RS-232 adaptor. The serial port connector(s) are on the rear panel of the packet controller. Purchase or build a cable that connects to the serial port connector of your computer and the RS-232 (or TTL) connector of your packet controller. The RS-232 cable should be wired for connection

INTRODUCTION

Software Requirements

from a computer (Data Termination Equipment - DTE) to a modem (Data Communication Equipment - DCE). See the **TECHNICAL REFERENCE MANUAL** for detailed computer interfacing information for specific packet controller models. PacComm supplies the mating connectors for all of the packet controller connectors. Complete cables for most popular computers are also available from PacComm.

Software Requirements

Any software that enables your computer to operate with an ordinary telephone modem should work with your packet controller. There are many sources of free or low cost communication software for use with a packet controller. Check your local packet radio Bulletin Board System (BBS), local telephone BBS, Public Domain software vendors in computer magazines and at Hamfests and Computer Shows.

Software specially written for packet data communications for both personal and business use for many popular computers is available for sale from PacComm.

Radio Interfacing

The packet controller 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. The packet controller's audio signals are fed directly into the microphone connector or similarly connected auxiliary jack of the radio, and the packet controller output is adjusted to give a proper modulation level. The receiver audio is taken from an auxiliary audio output or speaker jack on the radio and fed directly to the packet controller. Refer to the **TECHNICAL REFERENCE MANUAL** for detailed instructions for your model of packet controller.

Other Information

The information contained in this document has been carefully checked and is believed to be entirely reliable, however, no responsibility is assumed for inaccuracies. PacComm reserves the right to change products or documentation for any purpose including improvement of reliability and functionality without obligation to purchasers of previous equipment. PacComm assumes no 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.

OPERATING INSTRUCTIONS

Verifying Serial Port Operation

You may use either UPPER CASE (capital letters) or lower case (small letters) when you enter commands. The shortest acceptable form of each command will be shown in upper case in all the examples. In order to have your packet controller accept a command line, you must finish the line with a <CR>, or carriage return character. This won't be mentioned again in the examples below. Before you type the final <CR> of your command, you can correct typing mistakes or cancel the line completely.

This manual will show commands and packet controller responses in a box which roughly simulates a screen display. In examples showing input typed by the user together with the responses of the packet controller, the user's input will be boldfaced.

Whenever the packet controller accepts a command which changes a value, it displays the previous value to remind you of what you have done, and indicate that the value has been successfully changed. For example, if you type

```
cmd:Xflow OFF
```

you might see the display

```
Xflow was ON  
cmd:
```

If you type something your packet controller can't understand you will get the unrecognized command error message

```
?unknown command
```

If you get a command name correctly, but the arguments are wrong, you will see the message

```
?parameter
```

Connect the packet controller to the computer with a properly configured serial cable. Turn the computer on and start the terminal program. Follow the directions for the program you are using to set the computer's baud rate to 1200 bauds, 8 bits, no parity and 1 stop bit. The packet controller is delivered configured to operate at that speed and settings for the terminal unless otherwise indicated on the Errata Sheet. See the TECHNICAL REFERENCE MANUAL for details.

Turn on your packet controller. You should see the following display.

```

| A
PacComm [model] packet controller
AX.25 Level 2 Version 2.0
FEATURES:
Release X n.n.n Ym 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 packet controller. The line following **FEATURES:** will contain a list of optional firmware features included in this EPROM. See the Errata for values of N, nn, Y, m and xx for your model of packet controller.

The Command Mode prompt, **cmd:** will appear when the packet controller is in Command Mode and is ready to accept your instructions. If you see nothing, switch off your packet controller for a few seconds, then on again. If you still see nothing, verify your wiring and restart your terminal program.

Serial Port parameters

If messages from your packet controller appear garbled, with incorrectly displayed characters, you may need to change the parity and word length of either the computer terminal, the packet controller, or both. The most common parity and word length combinations are 8 bits, no parity (the packet controller default), and 7 bits, even parity. The packet controller 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

Some computers will 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 amount of delay is adjusted by **NULLs**, which sets a number of character-times for the delay.

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

```
cmd:EECCHOO
```

In this case your computer is echoing the characters you type, and the packet controller is also echoing them. Set **Echo OFF** to stop the packet controller's echos. If you later use your packet controller with a different computer, or different terminal program, and nothing is displayed when you type, set **Echo ON**.

If the packet controller's sign-on message lines appear to be typed over each other, you should set **AUtoLf ON**. If the packet controller's sign-on message appears to be double-spaced, you should set **AUtoLf OFF**. If the packet controller's sign-on message appears to be single-spaced, you have **AUtoLf** set correctly. If you change equipment you may have to set **AUtoLf ON** to restore the packet controller's automatic linefeeds.

If you are able to enter one command successfully and then begin receiving the ?unknown command message set **LFIgnore ON**, or set your computer terminal to send only a <CR> when the **ENTER** key is pressed.

If you make mistakes while entering commands to the packet controller, you can make corrections. To erase the last character you typed, enter a <BACKSPACE> character. The packet controller 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 (**Screenln**) defaults to 0, since most computers automatically insert a <CR> to format lines to the screen width. If **Screenln** is non-zero the packet controller will send an extra <CR> (or <CR><LF> if **AUtoLf** is **ON**) when the **Screenln** number of characters have sent. If your computer does not automatically break long lines, you will need to set **Screenln** to the width of your display. For example, using a TV set for a display, you would set **Screenln** to 23 or 40.

Some PacComm packet controller models have a **BAud** command for changing the terminal (and radio) data rates, while other models use switches or hardware jumpers.

OPERATING INSTRUCTIONS Beginning Packet Operation

See the packet controller **TECHNICAL REFERENCE MANUAL** to learn which method is employed on your packet controller. Complete information on the **BAud** command is included in the **Commands Listing**.

Beginning Packet Operation

This section will guide you through the basics of packet radio operation with the packet controller. Packet radio has a great deal of power and flexibility, and this section only scratches the surface of your packet station's capabilities. However, it contains the basic information required to begin operation.

First Steps

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

```
cmd:MYcall XXXXXX
was NOCALL
cmd:
```

Your **CALLSIGN** will be used by the packet controller as its "address." The packet controller responds with the previous value of the **MYcall** parameter, and gives a new **Command Mode** prompt. Now try typing just the command by itself:

```
cmd:MYcall
MYCALL XXXXXX
```

This verifies that the packet controller accepted your **CALLSIGN**. 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 **commands listings** in this manual and on the **COMMAND REFERENCE CARD**. For the sake of clarity, only the full command names are used in this manual.

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 try out your packet controller. Have another packet operator help you get started. Make sure that he will be close enough to ensure solid signals, with no FM "popcorn" noise. Try to have both stations in the same room and operate on low power or into dummy loads.

OPERATING INSTRUCTIONS Beginning Packet Operation

Turn on your computer and the packet controller. Be sure you have adjusted your packet controller and radio according to the methods described in **TECHNICAL REFERENCE MANUAL**. When the other station transmits, the DCD LED on your packet controller 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 packet controller on the air. Remember to set your radio squelch control so no noise is being heard from the radio except during packet transmissions.

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.

Starting the Connection

You are ready to initiate a connection. For the sake of example, we will continue to use N2WX in place of your call sign, and we will use K4GFG for the other station’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. The other station will see the message

```
*** CONNECTED to N2WX
```

and he will also be in Converse Mode. If you have trouble connecting, make sure your microphone drive level is set properly. It may be helpful to have an experienced packet operator listen to your transmissions.

If you monitor the radio transmit indicators and listen to the speaker audio from the two radios, you will have a better idea of what is happening. Your 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 the other station will hear the “brrrraaaap” sound on his speaker. As your message is displayed on his computer screen, his radio will be keyed for an even shorter time and you will hear a “brrraap” on your speaker. This is the ACK, or packet acknowledgment coming back. Your packet controller takes note that the packet was received correctly, but nothing is displayed on your screen.

OPERATING INSTRUCTIONS Beginning Packet Operation

In the example above, your packet controller entered Converse Mode automatically after the connect took place. 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 packet controller doesn't echo the **<CTRL-C>**, but you should immediately see a Command Mode prompt. To terminate the connection, you must type the **Disconnect** command. The packet controller will transmit packets terminating the conversation and notify you when the disconnect is complete:

```
cmd:Disconnect
*** 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.

In Case of Difficulty

If there is difficulty getting a connection established, you can try the following procedure. Both stations should set **Monitor ON**, then enter Converse Mode and send some packets. Each station should display the 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 packet controller. Set **TXDelay 64** for a long delay. If this solves the problem, try shorter and shorter delays and use the smallest value that works consistently.

Digital Repeating (Digipeating)

You may wish to have a connection with another packet station

OPERATING INSTRUCTIONS Beginning Packet Operation

that is beyond your direct radio range. If a third packet station is on the air, that station can relay your packets. You set up the packet routing when you initiate the connection. Your packet controller will then automatically include the routing information in the packets it sends.

The diagram below shows a situation in which digipeating is useful.

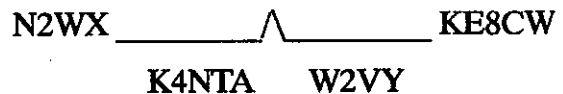


You are station N2WX, and you want to have a packet connection with KE8CW, who is not in direct range. However there is a station AD7I, which is in range of both you and KE8CW.

You direct the packet controller 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:



You can contact KE8CW by going through K4NTA and KV7D. This time you issue the connect command like this:

```
cmd:Connect KE8CW Via K4NTA, W2VY
```

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. Digipeating may be disabled by the command **DIGIpeat OFF**. The command **MYAlias** will allow setting an alternate callsign for your packet controller for digipeater identification.

When an alias callsign is set, the controller will digipeat frames addressed to either MY call or MYAlias callsigns, but will accept connections only to MY call.

Now that you are on the air, you can try out the packet controller'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 and issue a new connect command:

OPERATING INSTRUCTIONS Beginning Packet Operation

```
< CTRL-C >  
cmd:Disconnect  
*** DISCONNECTED  
cmd:Connect N2WX VIA K4GFG
```

(Remember to use actual callsigns, not the ones in the examples.). You are initiating a connect to yourself. You transmit packets to the other stations packet controller, 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. Monitor the radio transmit indicators and listen to the speaker audio. See if you can follow the packets and the acknowledgements back and forth. The other operator won't see anything displayed on his computer and his packet controller's state won't be affected at all by your connection. In fact, the other station could carry on a separate conversation completely independently.

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 packet controller does not get a response to its first connect packet, it will try again. You can control the number of attempts the packet controller will make with the command **REtry**. The default number of retry attempts is 10. If the packet controller 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 packet controller 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 packet controller will automatically disconnect and send the same message shown above.

The number of tries may be determined at any time by entering the command **TRies**.

OPERATING INSTRUCTIONS Beginning Packet Operation

The automatic disconnect feature keeps a packet controller from indefinitely retransmitting a packet and tying up the channel under hopeless conditions. If you are operating under special conditions you can set **REtry 0** to disable automatic disconnects (the retry limit is never reached).

If unacknowledged data packets existed at the time of the disconnection, the packet controller will send all of these data packets as Unnumbered Information (UI) frames. To prevent the dumping of these frames over the air, set the command **TXUIfram OFF**.

If you have a connection established but wish to change the path being used (the sequence of stations listed after the **VIA** in the connect command) use the **REConnect** command. This command is used **WHILE ALREADY CONNECTED**. For example, if a connection exists to W2VY via N2DSY and you wish to change the path to be W2VY via KD6TH, N2DSY, issue the command

```
cmd:REConnect W2VY via KD6TH, N2DSY.
```

Monitoring Channel Activity

The packet controller can allow you to monitor packet activity on the channel. Monitoring is enabled or disabled by the **Monitor** command.

```
cmd:Monitor ON  
cmd:was OFF
```

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:

```
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 packet controller 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**.

If you want to monitor without the display of any callsigns, set **ADdrdisp OFF**. This would cause the last example above to be displayed as:

```

Hello Bill!

```

The **ADdrdisp** command is mainly useful to monitor the packets from a single station. Use the commands **LCalls** and **BUdlist** to limit monitored packets to those of one station. If monitored frames seem to be double spaced when using **ADdrdisp**, set **CRAfter OFF**.

To display a list of stations heard since the last time your packet controller was powered up, type the command **MHeard**. The last eighteen packet stations heard by your packet controller 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.

The Heard Log has an expanded format which includes the type of station heard. If the received frames can be identified properly, the term **NODE**, **BBS**, or **PMS** will be added to the Heard Log entry following the callsign for that station.

Nodes are identified by the transmission of a level 3 type of frame. **BBS** and **PMS** stations are identified by an identification field of the format [XXX-xxxx], e.g. [PMS-3.0-C\$]. Some **BBS** programs do not conform to this format and will not be identified in the Heard Log. Nodes may be misidentified when handling **BBS** or **PMS** messages.

```

cmd:mh
W4DPH-15   BBS      07/30/90  17:31:12
K0ZXF     BBS      07/30/90  17:31:11
W8DUV*    BBS      07/30/90  17:26:14
CLW5      NODE     07/30/90  16:55:00
DAB5      NODE     07/30/90  16:50:35
K4HFL                    07/30/90  16:50:09
W4MUA-1   BBS      07/30/90  16:46:23
W8DUV-15  BBS      07/30/90  16:43:46

```

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**.

Special Characters

The packet controller recognizes a number of special characters for input editing, flow control, and other control functions. You

may change any of these special characters. 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 the other 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 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 the Commands and Messages Section for the command that sets that character.

You can enter the code for a character in either hexadecimal (HEX, base 16) or decimal notation. The packet controller displays character codes in hex. A number of hex notation is indicated by beginning the number with a \$. The "digits" of a hex number represent multiples of the 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. A table of ASCII codes for control characters appears below.

Special Input Characters

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**. 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 packet controller displays the previous value in hexadecimal notation (hex). All of the special characters described below can be changed in the same way as **COMmand**.

If you set **DELeTe ON**, you can erase characters by typing the **<DELETE>** character; setting **DELeTe OFF** returns to using **<BACKSPACE>**. **<BACKSPACE>** is more commonly used than **<DELETE>** by personal computers. If you aren't sure whether your rubout key produces **<DELETE>** or **<BACKSPACE>** characters, try both settings and see which works.

ASCII Codes for Control Characters

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

When you rub out a mis-typed character, the packet controller will attempt to correct the screen display. If your display doesn't look right after you rub out a character, try setting **BKondel OFF**. The packet controller 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 want to cancel a whole line rather than rubbing out the characters one at a time, type **<CTRL-X>**. The controller will display a **<BACKSLASH>** followed by **<CR>** and (in Command Mode) a new prompt:

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

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 **BKondel** is set **OFF**. Type **<CTRL-R>** to have the controller retype the line you are entering.

cmd:Connect KB7\\WA7 < CTRL-R > \	[Call sign
cmd:Connect WA7	mis-typed.]

Here the user mis-typed the first three characters of the call sign and rubbed them out. The packet controller 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 **REDisplay**.

If your packet controller displays information faster than you can read it before it scrolls off the screen, you may halt the display by typing <CTRL-S>. To resume scrolling, enter <CTRL-Q>. These characters can be changed to any ASCII character by the commands **STOp** and **STArT**.

You can include any character in a packet, including any special character, 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>, it is sent all at once, but maintains the <CR> in the text. The pass character can be changed to any ASCII character by the command **PASs**.

Operating Modes

The packet controller has four operating modes. Two of these modes, Command Mode and Converse Mode, were discussed above. 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. The other data-transfer mode, KISS, is a specialized data-transfer mode for use for MicroSat satellite access and the TCP/IP protocol.

Command Mode

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

The packet controller will be in Command Mode when power is turned on. After a power-off, power-on sequence, or the restart command, all operating parameters of the packet con-

OPERATING INSTRUCTIONS

Operating Modes

troller are re-initialized to the parameter stored in battery backed-up RAM (bbRAM). After the **RESET** command is issued all operating parameters are reset to the default values stored in EPROM. You probably won't use this command often. The packet controller will automatically reload the bbRAM from EPROM at power up if it finds that the data is bad.

The following commands set special characters which are active in Command Mode.

CANline	Cancel current line
CANPac	Cancel output (Command Mode only)
DELete	Character deletion
PASs	Insert following special character
REDisplay	Re-display current line
STArt, STOp	User's flow control (sent from computer)
XOff, XON	Packet unit flow control (sent to computer)

The following commands enable display features which are active in Command Mode.

AUtoLf	Add <LF> after <CR> (data to term)
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
ScreenIn	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 **CONVerse** (or the single-letter command "K") to enter Converse Mode or the command **TRansparent** to enter Transparent Mode, and the packet controller will immediately enter the specified mode. The packet controller will automatically enter a data-transfer mode when a connection is made. You specify the data-transfer mode for automatic entry with the command **CONMode**:

```
cmd:CONMode Transparent
```

```
cmd:CONMode Converse
```

specify Transparent Mode, and Converse Mode respectively.

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 packet controller accepts, you will enter data-transfer mode when the packet controller sends the connect acknowledgment (ACK) and displays the message:

```
*** CONNECTED TO < callsign > .
```

If you initiate the connection with the **Connect** command, you can control the timing of the mode change using the commands **NEwmode** and **NOmode**. If **NOmode** is OFF and **NEwmode** is OFF, the mode will change when the connect ACK is received and the ***** CONNECTED TO: callsign** message is displayed. If **NOmode** is OFF and **NEwmode** is ON, you will enter data-transfer mode immediately, without waiting for a successful connection. Any text sent to the packet controller 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.

If the command **NOmode** is ON, then no mode changes will be made automatically. This is an ideal way to insure control by software in a host computer.

KISS Mode

The KISS mode is normally used for placing the packet controller in a simple data transfer mode controlled by the attached host computer. This is used when using Transmission Control Protocol/Internet Protocol (TCP/IP), or other communication software which interfaces with KISS, such as that used for accessing the MicroSat packet satellites. There is no value in entering the KISS mode unless such operation is contemplated.

To enter the KISS mode, enter the command

```
cmd:KISs ON
```

followed by the command

```
cmd:RESTART
```

The packet controller will stay in KISS mode despite use of the **RESET** command or turning the controller OFF. This 'permanent' activation of the KISS mode allows the user to access the KISS function reliably despite power outages or other problems. The battery backed RAM parameters for conventional packet operation are retained for later use.

To exit the KISS mode, the TCP/IP computer must issue the

command **PARAM AX0 255**, or the packet controller must be given a hard reset by removing the backup power from the RAM, or you can do the following from an MS-DOS PC keyboard with most communication programs: 1) Hold down the ALT key, 2) Press keys 1 9 2 on the numeric keypad, 3) Release the ALT key, 4) Hold down the ALT key again, 5) Press keys 2 5 5 on the numeric keypad, and 6) Release the ALT key.

Converse Mode

The data mode used often for ordinary keyboard conversation 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.

The following commands set special characters which are active in Converse Mode.

CANline	Cancel current line
CANPac	Cancel current packet
COMmand	Command Mode entry
DElete	Character deletion
MFilter	Monitored characters to be filtered
PASs	Insert following special character
REDisplay	Re-display current line
SEndpac	Send current packet
STArt, STOp	User's flow control (sent to packet unit)
XOff, XON	Packet unit flow control (sent to computer)

The following commands enable display features which are active in Converse Mode.

8bitconv	Retain all serial port bits in converse mode
AUtoLF	Add <LF> after <CR>
BKondel	Echo after character deletion

OPERATING INSTRUCTIONS

Operating Modes

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
ScreenIn	Automatic < CR > insertion

Transparent Mode

Converse Mode will often work well for computer data transfer, but files such as a .EXE 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. If you transfer such files you will have to use Transparent Mode.

Transparent Mode recognizes no special characters — everything you type (or everything your computer sends to the packet controller) is sent over the radio exactly as it was received by the packet controller. 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 **PACTime**.

The display characteristics of the packet controller 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 packet controller is returned to Command Mode. Most of the link status messages that appear as the packet controller 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, you 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 these CMdtime intervals (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 procedure.

You can also enter Command Mode from Transparent Mode by sending a BREAK signal. If the computer sending data to the packet controller inadvertently sends a data signal interpreted as a BREAK, set **BR**eaK OFF to disable the packet controller's response to the signal.

Flow Control

The packet controller'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 because you are using a faster serial port baud rate than the radio data rate, or radio data transmission may have slowed down because of noise or other users on the channel, or the person or computer at the other end may have stopped output from that packet controller. The packet controller 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 packet controller will signal the computer to start sending data again.

There are two methods of providing flow control which are supported by the packet controller. 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 Request To Send (RTS) 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 RTS and CTS lines appear at the connector, they may not be used unless the software reads the state of the CTS line. If you find that the packet controller 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 RTS/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. The commands **XON** and **XOff** set the characters which will be sent to the terminal by the packet controller, and the commands **STArt** and **STOp** set the characters to be sent to the packet controller by the terminal. Your computer may receive as many as 4 characters from the packet controller 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 packet controller 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 packet controller 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 **RTS/CTS** flow control, you may enable the **XON** and **XOff** characters (the commands from the packet controller to the terminal) by setting **TXFlow ON** and **Xflow ON**. The **STArt** and **STOp** characters (the commands to the packet controller 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. **RTS** 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 packet controller to stop sending, since some characters may already be "enroute" through serial I/O chips. Refer to the **TECHNICAL REFERENCE MANUAL** 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 which keeps the packet controller from interrupting you with incoming packets. As soon as you type the first character of a line, the packet controller 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

OPERATING INSTRUCTIONS

Station Identification

send-packet character to mark the end of a packet, or until you erase or re-display the line you have started. This is most helpful if you are not using a split screen communication program.

Station Identification

Your station identification (callsign) is set with the command **MYcall**, as described previously. If you will have more than one station on the air operating with the same callsign, 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 callsign with a dash:

```
cmd:MYcall K3AAX-3
```

If you don't specify the **SSID** extension, it will be 0, however the packet controller won't explicitly display **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 packet controller can send an automatic identification packet every 9-1/2 minutes. You can enable this with the command **HIId 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 packet controller will only send identification packets if it has been digipeating.

Automatic Operations

Unattended (automatic) operation is legal in the USA on frequencies above 30 MHz. Check the latest FCC rulebook or the governing authority in your country.

Unattended Connections

Your station will normally accept a connect request from another station if it isn't already connected or if multiple connections are enabled. You can disable this capability by setting **CONOk OFF**. If a connect request is received when **CONOk** is **OFF**, the packet controller will display the message:

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

OPERATING INSTRUCTIONS

Automatic Operations

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 packet controller will display

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

If you want to have a special 'connect 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: (CTRL-V)

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

In order for this message to be sent, **CONOk** must be ON so that the connection takes place (default), and **CMSg** ON. You can cause your packet controller to issue a disconnect request as soon as the connect message is acknowledged by issuing the command **MSGDisc** ON.

NOTE: Turn **MSGDisc** off when you resume in-person operation of your station to prevent premature disconnects.

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

Beaconing

You can have your station periodically send an automatic message by enabling "beacons." 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 30 minute intervals, give the command

```
cmd:Beacon EVERY 180
```

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.

Use the Beacon feature with caution since much needless interference may be caused to other users of the channel by careless beaconing.

Connect Alarm

The command **CBell ON** will cause a 'bell' character (^G, \$07) to be sent to the terminal each time a connection is established.

Packet Formatting

The maximum length of a packet is determined by the command **Pacten**. If you type more than the maximum number of characters without entering a send-packet character, the packet controller 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. (Older controllers may not 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

The packet controller implements AX.25 Level 2 Version 2 protocol, a set of rules for formatting messages to other packet controllers. This may be set to Version 1 with the command **Ax25l2v2 OFF**. Digipeating may not be successful if some packet controllers are running Version 1.0 and some are running Version 2.0.

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

```
cmd:Unprotocol 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 Transmit Timing

Radio equipment requires a time delay for switching between transmit and receive modes. If the packet controller 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 packet controller is keying the transmitter but not sending data, it will transmit a synchronizing signal (flags).

If you are transmitting packets through a voice 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 packet controller will assume is required for the repeater to drop. If the packet controller 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.

Packet Timing

The AX.25 protocol provides for retransmitting packets if no acknowledgment is heard from the intended destination station within a certain period of time. A packet might not be acknowledged due to channel noise or “collision” with another packet transmission, and since there may be other stations on the channel, the receiving station may not be able to acknowledge the received packet immediately. The time lapse before the originating station retransmits the packet is set by the command **FRack** (frame acknowledge time). Do not set **FRack** to a small value as this will cause unnecessary retries, especially on HF.

The maximum number of retransmissions before the originating station terminates the connection is set by the command **REtry**. The maximum number of transmissions of a packet is **REtry** + 1, since the initial transmission does not count as a retransmission. Setting **REtry** to 0 specifies an infinite number of retries.

Acknowledgments of digipeated packets are made from end to end, and digipeaters do not acknowledge the packets they relay. If there are several intermediate relays, the chance of either the original packet or the acknowledgment to be lost increases drastically. To reduce this problem, an automatic wait time can be imposed on any station not transmitting a digipeated packet. This usually gives the digipeater a clear chance at the channel. The wait time is set by the command **DWait**, which specifies 40 ms intervals. If no digipeating is being done in the local area, this parameter can be set to 0, but in any event it should be set to the same value by all members of a local packet group.

In order to avoid unnecessary packet retries, the packet controller implements a collision-avoidance strategy which applies to all packets except those being digipeated. On the second and subsequent transmissions of a particular packet, the packet controller waits an additional random time after detecting a clear channel before beginning transmission. This prevents repeated collisions of transmissions by the same two stations. The random time is a multiple (0-15) of the **TXDelay** time. The interval, in milliseconds, between the packet controller detecting carrier-drop and beginning to transmit is

- Wait time = $DWait * 10$

for the first transmission of a packet. For subsequent transmissions of the same packet the interval is

- Wait time = $DWait * 10 + (r * TXDelay) * 10$

where r is a random number from 0 to 15. Thus, if your packet controller is forced to retransmit packets, you will occasionally hear a fairly long delay before transmission begins.

Multiple packets may be transmitted before waiting for an acknowledgment. This permits more efficient channel use when large amounts of data are being transferred. The maximum number of packets which the packet controller will send before waiting for acknowledgment is specified by the command **MAXframe**. **MAXframe** in combination with **Paclen**, which sets the maximum number of characters in a packet, determines how much information can be sent in a single transmission.

Alternate Transmit Timing

An alternate method of transmit timing called P-Persistence may be implemented by the command **PPersist ON**. (This disables the retry timing using **FRACK**.) This timing method is most effective if all users on a channel make use of it.

Persist uses the parameters **PEpersist** and **SLOttime**. When a frame is queued for transmission and the **DCD** is not active, a

transmit/wait decision will be made based comparing a random number between 0 and 255 to the value of **Persist**.

If the random number is less than the **Persist** value, the channel is seized. If not, then a delay of **SLottime** is timed, and the random number/**Persist** calculation is repeated. Thus large values for **Persist** and small values for **SLottime** produce the most aggressive channel access, and small values for **Persist** and large values for **SLottime** produce the most civilized channel access.

Special Protocol Times

You can set up a connection timeout with the command **CHEck**, which specifies a time in multiples of 10-second intervals. This function prevents your packet controller from getting stuck in a connection when the other station disappears for longer than the specified time. The packet controller uses this time somewhat differently depending on the setting of **Ax25I2v2**.

The command **RESptime** sets a delay between the receipt of a packet and the transmission of the acknowledgment packet. This delay is used to prevent collision between an acknowledgment and another packet from the sending station. This is primarily necessary during file transfers; otherwise the delay is best set to 0 (default). During file transfers the stations receiving the file should set **RESptime** to 10 or 12.

The timing of packet transmission in Transparent Mode is determined by the command **PACTime**. You can choose the way packet transmission is timed. If you are typing input to a remote computer it is usually best to have packets transmitted at regular intervals. If your computer is operating a remote host or bulletin board program you should send packets after an interval with no further input from the computer. You can enable the use of **PACTime** in Converse Mode with the command **CPactime**.

Monitor Functions

Monitoring of packets not addressed to your station is enabled by the command **Monitor ON**, and separate monitor functions are individually enabled.

The command **MAI OFF** causes the packet controller monitor function 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 **MAI** 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.

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 allows every correctly received frame to be displayed. For example:

```
WA7GXD>KV7D <I C S0 R0>: Hi Dan,
WA7GXD>KV7D <I C P S1 R0>: been on HF ?
KV7D>WA7GXD <RR R F R2>
KV7D>WA7GXD <I C P S1 R2>: no, 10 meters is dead.
```

The abbreviations are explained in the protocol section.

You can cause the packet controller 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.

The command **PId** ON will eliminate the display of received frames whose Protocol Identifier (PID) is not \$F0. This should prevent most unwanted 'NODE' transmissions.

"Health" Features

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

Twenty five counters, all of them 16 bits wide, are provided to monitor a number of parameters in the packet controller. 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**.

●ASYFRERR:

Asynchronous Framing Error. Incremented when a receive framing error is detected on the serial port IC.

●ASYQOVER:

Asynchronous Overflow Error. Incremented each time a character was received by the serial port IC but not accepted by the firmware.

●ASYRXOVR:

Asynchronous Receive Overflow Error. Increases when the software does not service the asynchronous receiver in time. Indicates data from the user to the packet controller is being dropped.

●BBFAILED:

Battery Backed RSM failures. Counts number of times bbRAM checksum was in error.

●DIGISENT:

Duplicate Frames Sent. Each frame duplicated by this packet controller causes the counter to increase.

●HOVRERR:

HDLC Overrun Error. Increases when HDLC receiver is not serviced rapidly enough and data is lost.

●HUNDRERR:

HDLC Underrun Error. Increases when the HDLC transmitter is not serviced rapidly enough and frames are aborted.

●PMSCONN:

PMS Connections. Increments once for each connection established with MYPCALL.

●RCVDFRMR:

Received Frame Rejections. Increases when Frame reject frames are received from a connected station.

●RCVDIFRA:

Received Frames. Increases for each reception of an I-frame from a connectee.

●RCVDREJ:

Received Reject Frames. Increases for each reception of an REJECT frame from a connected station.

●RCVDSABM:

Received SABM Frames. Each received SABM frame addressed to the packet controller causes this counter to be increased by one.

●RCVDRNR:

Received RNR frames. Counts the number of received Receiver Not Ready (RNR) frames. Incremented each time the connected station sends an RNR because it cannot take any more frames.

●RXABORT:

Received Aborts. Counts the number of received Abort Signals. Abort signals are generated by the HDLC chip, not the AX.25 firmware. Abort means, in effect, a cancel frame signal.

●RXCOUNT:

Received frame count. Increases when any frame is received with good CRC (or any CRC if PASSAll is turned on).

●RXERRORS:

Received Errors. 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.

●RXLENERR:

Received Length Error. Receiver detected a frame less than 15 bytes in length.

●RXRESYNC:

Receive Resynchronization counter. Increments each time a difference between the received packet length and the buffer data length has been detected, causing buffer resynchronization.

●SENTFRMR:

Frame Rejects Sent. Increments for each Frame Reject frame transmitted.

●SENTIFRA:

Information Frames Sent. Increases by one for each I frame sent.

●SENTREJ:

The number of Frame Rejects Sent. Whenever a REJect frame is transmitted, this counter is incremented.

●SENTRNR:

The number of Receive Not Ready (RNR) frames sent. RNR frames are sent when the packet controller is unable to accept any more frames, usually due to flow control by the computer.

●TXCOUNT:

Frames Transmitted Counter. Incremented whenever a frame is correctly transmitted.

•TXQOVFLW:

Transmit Queue Overflow. Counts how many times frames were discarded because the outgoing frame queue was too small.

•TXTMO:

Transmitter Timeouts. Counts the number of HDLC transmitter timeouts.

Clock Operations

The packet controller clock is maintained in software by the Z-80 microprocessor.

You enable the packet controller'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 date and time will be lost each time the packet controller is powered off. The command **CLKadj** is used to 'tweak' the software clock toward better accuracy. If the clock loses time, increase the value of the **CLKadj** parameter, *n*. Large values have a small effect on the time, while small values of *n* have a large effect on the time.

Since the clock accuracy varies with the amount of activity on the radio channel, it may be impossible to get complete clock accuracy.

NOTE: Some PacComm packet controllers have an optional hardware clock which retains the date and time even if the packet controller is powered off.

The format of date and time display is controlled by the commands **DAYusa** and **AMonth**. **DAYusa** allows selection of a US or European date format and **AMonth** allows selection of alphabetic or numeric month format. The **AMonth** command does not affect the date format of the Personal Message System.

Time Stamping

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**.

Special Operations

If you are operating a full-duplex radio station (simultaneous transmit and receive), you should set **FULLdup ON**. The packet controller is always electrically capable of full duplex operation, but this parameter causes the packet controller will ignore carrier detect status.

For satellite and other full duplex modes of operation you should set **DWait** to 0. The random wait before retry transmissions can be disabled by setting **TXDelay** 0 and **AXHang** 0 and using **AXDelay** to set the required keyup delay.

Computer Operations

The **BBmsgs** command is particularly useful when the packet controller is used with a host computer. When ON, the command causes all packet controller messages to either be suppressed or to begin on a fresh line.

The **RXblock** command should be used when the packet controller 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 packet controller-generated messages.

If unacknowledged data packets exist at the time of a disconnection, the packet controller will send all of these data packets as Unnumbered Information (UI) frames. To prevent the dumping of these frames over the air, set the command **TXUIfram** to OFF.

The **STATus** command is useful for allowing a host computer to obtain information about whether a connection exists, whether there are any unacknowledged frames, and whether a pending connection or disconnection exists.

HF Operation

Some PacComm packet controllers are equipped with both VHF and HF optimized modems. See the TECHNICAL REFERENCE MANUAL for a description of your controller's capabilities. Dual-modem boards have several additional commands to support HF operation.

The **POrt** command selects which radio port is in use. The **POrt** command causes most parameters except **callsign** to be changed to correspond with the use of the selected port, including the radio baud rate, modem tones, and link timing parameters. If the **POrt** command is given without an argument, the packet controller responds with the number of the currently selected port. If **POrt** is given followed by a port number such as:

```
cmd: PO 2
```

the packet controller switches to the specified port and responds with the number of the previously selected port.

```
was 1
```

Only packet controllers termed 'DUAL PORT' in the TECHNICAL REFERENCE MANUAL are capable of simultaneous operation of the radio ports. Dual-radio or dual-modem models disconnect the port not currently selected.

Some PacComm packet controllers allow selection of the carrier detect signal (DCD) from either the modem circuit (hardware) or a software algorithm (software). The command **Softdcd ON** selects the software type of DCD, while **Softdcd OFF** selects the hardware signal for DCD. Consult the TECHNICAL REFERENCE MANUAL to determine if this command applies.

Multiple Connections

With multiple connections, you may establish simultaneous connections with several stations, and the packet controller will keep the data to and from each connection segregated. The packet controller defaults the multi-connect-related parameters to the following 'single connection' values:

CONPerm	OFF
LCStream	OFF
STREAMCALI	OFF
STREAMDbI	OFF
STReamswitch	
USers	1

The key to enabling multiple connections is to set **USers** to a value greater than 1. If **USers** is not 1, the packet controller will allow multiple connections (as many as the value of **USers**) to your station 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 while 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 **STReamswitch** character. When operating a single user Mail-Box (BBS) system or other host computer application, it is highly recommended that **STReamsw(itch)** be set to \$00 (disabled).

Although not foolproof, enabling **STREAMDbI** may also help in sorting out **STReamswitch** characters included in the

OPERATING INSTRUCTIONS Personal Message System

received data from a valid stream switches generated by your packet controller.

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 packet controller with the stream identification.

When in command mode the stream may be switched by entering the **STReamsw(itch)** character (default "|"), followed by a stream identifier ("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 **STReamsw(itch)** 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. Note that the stream identifier must be in upper case (A through J) unless **LCStream** is set to ON.

When the packet controller firmware contains the Personal Message System (PMS), stream K is used for PMS functions and is displayed as part of the **CStatus** command display.

Personal Message System

Warning: The firmware contained in all PacComm EPROMs is copyrighted by PacComm and may not be copied for any purpose.

The PacComm Personal Message System (PMS) is a self-contained message system which operates in all of PacComm's packet controllers. The PMS is contained in an EPROM along with the regular packet firmware. It is a standard feature on most PacComm packet controllers. The PMS allows the packet controller to support all standard packet functions while simultaneously allowing messages to be entered or read by an over-the-air user. PMS requires 32k of RAM to operate. Approximately 13-15k of battery backed RAM is allocated for messages. Messages are retained when the packet controller is powered off. Some PacComm controllers support 512k of RAM for message storage!

The PMS features complete compatibility with current BBS message standards. Message headers are optionally generated to allow tracking of the message in the BBS system. Message headers may be edited to allow message readdressing. To im-

OPERATING INSTRUCTIONS Personal Message System

provide operator convenience the PMS supports forwarding from, and reverse forwarding to the local (home) BBS. Automatic forwarding by the PMS is NOT supported to reduce channel congestion. If the PMS is installed on a PacComm controller which is equipped with a printer port, messages may be printed directly to the printer at keyboard command.

The PMS supports the Bulletin ID (BID). A BID may be entered following the TO callsign field when using the SEND (or SP or ST) command. The dollar sign character must be present, but the following field may contain up to 28 characters of any type. In standard practice the Dollar sign is followed by a BID number, an Underbar character and a callsign. The line must terminate with a carriage return. An example of the BID format:

```
SP N4UQQ @ K0ZXF $53 _N4UQR
```

The BID does not appear in the message display header in response to the LIST command but does appear in the presentation of each individual message in the following format:

Stat :
Posted :
To :
BID :
Subject:

PMS Setup Commands

The packet controller operator is referred to as the PMS SYSOP (system operator). The commands which the Sysop uses to configure the PMS are:

- 3rdparty
- AUTOFwd
- CLKSet
- HOMebbs
- KILONFWD
- LOGonmsg
- MSGHdr
- MYPcall
- NODEtext
- PMs
- REMSysop
- STExt

To enable the Personal Message System, set **PMs ON** and enter a callsign in **MYPcall**. The callsign must NOT be identical to

OPERATING INSTRUCTIONS Personal Message System

MYcall. Check with other PMS operators in your area to determine what SSID is normally used to identify personal BBSs.

If the Sysop desires to allow messages to and from other stations to be left on his PMS, set **3rdparty ON**. Setting **3rdparty OFF** will cause users to be prevented from leaving messages to other than **ALL** or the PMS callsign. The Sysop may address messages to any callsign regardless of the setting of **3rdparty**.

HOMebbs is used to set the callsign of the local BBS which forwards mail to your PMS. The PMS uses BBS forwarding protocol when sending messages to the **HOMebbs**, but uses a user protocol for all other message transfers.

CLKset ON allows the **HOMebbs** to automatically set the packet controller to the correct time during a forwarding session.

AUTOFwd ON causes any messages entered by the sysop to be automatically marked for forwarding.

KILONFWD ON causes all messages which have been reverse forwarded to be deleted from the PMS to preserve memory capacity for incoming messages.

The **NODetext** command accepts a connect string for use by the **FNPMS** command. This connect string is used by the local node to connect to the desired forwarding station.

If the Sysop does not desire the standard PMS logon message to be sent, then set **LOGonmsg OFF**. A custom logon message of up to 80 characters may be entered in **STExt** and will be sent each time a station connects to the PMS. If **LOGonmsg** is **ON** and **STExt** contains a message, both will be sent. If **LOGonmsg** is **OFF** and **STExt** is blank, neither message will be sent.

If you are using the reverse forwarding capability of the PMS, set **MSGHdr ON** to insert a message header in outgoing messages.

The **REMSysop ON** command causes the PMS to allow the same privileges (read or kill any message) to the sysop's callsign over the air as well as from the keyboard.

See the individual command descriptions in the **PACKET CONTROLLER COMMANDS** chapter for further information.

PMS Operator Commands

The commands used by the Sysop to operate the PMS are:

EDithdr
FOrward
FPms

FNPms
KIll
List
MIne
PPRint
Read
SEND
SP
SR
ST

Note that some of these commands take a different form than used by an over-the-air PMS user (below).

The Sysop may list all messages currently stored in the PMS by entering the **List** command at his computer keyboard. If only messages addressed to or from the sysop's PMS station (MYPcall) are of interest, use the **MIne** command. Messages may be read by entering **Read n**, where **n** is the message number of interest. If the packet controller is equipped with a printer port, then messages may be sent direct to the printer attached to the packet controller with the command **PPRint n**. Messages are deleted by the command **KIll n**.

If any station has left a message in the PMS since the last time the Sysop accessed it, the PMS LED (STA LED on some models) will blink slowly to indicate that a message is waiting.

To send a message, use the command **SEND**. If the message is to be sent via the amateur packet BBS system, use the form of the command:

SEND callsign @ bbs-callsign \$optional BID.

The command **SP** is an extension of the **SEND** command which marks the message as personal or private.

The command **ST** marks the message as official National Traffic System (NTS) traffic and the text of the message should be entered in NTS format.

The command **SR n** creates a reply message addressed to the sender of message **n**.

If an error is made in entering the callsign header, or if a message is to be readdressed, use the **EDItHdr** command to change the **TO** or **@** callsigns.

To mark a message for automatic reverse forwarding, use the command **FORward n**. If a message is already marked for for-

OPERATING INSTRUCTIONS Personal Message System

warding and is not to be forwarded, the command **FOrward n** will 'unmark' the message.

Marking messages for forwarding causes no forwarding to take place unless the **HOMebbs** connects to the PMS or the operator uses either the **FPms** or **FNPms** command. The **FPms** is used when the station is reached via a direct connection or via a digipeater string. The **FNpms** is used when a node exists in the forwarding path.

See the individual command descriptions in the **PACKET CONTROLLER COMMANDS** chapter for further information.

PMS User Commands

The commands used by the over-the-air user of PMS are listed below.:

(B)ye
(H)elp
(J)log
(K)ill
(KM)
(L)ist
(M)ine
(R)ead
(S)end
(SR)
(V)ersion

All user commands consist of one or two letters which are shown in parenthesis. Use of more than the designated letters of the command will result in the message:

```
[PMS CMD ERR]
```

To use the PMS, the user simply connects to the PMS callsign. If PMS is ON and no other station is using the PMS, the following messages will be sent after the connection is established:

```
*** CONNECTED TO (PMS callsign)
Logged on to (PMS Callsign)'s Personal Message System
This line may contain an optional custom PMS message

CMD(B/H/J/K/KM/L/M/R/S/SR/V/?)>
```

See the individual command descriptions in **PACKET CONTROLLER COMMANDS** chapter for further information.

PACKET RADIO PROTOCOL

The following material is intended to supply an overview of the protocol used to transmit data by the packet controller. It 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 PacComm packet controller 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 PacComm packet controller 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 cable or modem tones on phone or RF links

The physical layer of the packet controller is described in the TECHNICAL REFERENCE MANUAL. It is compatible with the various packet controllers currently available. 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 PacComm packet controller, 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 packet controller 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. Version 1 is supported for compatibility with older packet controllers only.

HDLC Frames



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 circuitry used in the packet controller encodes the transmitted data 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.

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 packet controllers, 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 packet controller is not connected to another packet controller, e.g., CQ and QST activities.

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 interpretation 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 for packets not using Version 2.0.
- **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 its 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. The MAXframe parameter has a ceiling of 7: no more than seven frames can be "in flight" (transmitted but unacknowledged) at once.

AX.25 CONTROL CODES AND MCOM DISPLAY CODES

Code	Abrv	Frame Type	Code	Definition
x1	RR	Receive Rdy	C	SABM connect
x5	RNR	Receive Not Rdy	D	DISC disconnect
x9	REJ	Reject	UA	Unnumbered ACK
03	UI	Unnumbered info	DM	Disconnected mode
0F	DM	Disconnected mode	FRMR	Frame reject
2F	SABM	Connect request	U I	Unnumbered info Beacons, Id, etc.
43	DISC	Disconnect request	RR	Explicit ACK, R field
63	UA	Unnumbered Ack	RNR	Rcvr not rdy, R field
87	FRMR	Frame reject	REJ	Reject, R field
even	I	Any frame ending in an even number (\$A, \$C, and \$E) is an info frame.	I	Sequenced info frms, 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 packet controller is disconnected. Sent in response to an SABM whenever the packet controller can't connect to the requesting user, e.g., if the packet controller is already connected or 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

The following discussions mention timing parameters which are set by various commands.

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 packet controller 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 packet controller 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 packet controller 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 packet controller's response is based on the assumption of a collision.

After transmitting a packet, the packet controller 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 packet controllers accepting data at high speed from the asynchronous port from colliding with acknowledgment frames when fewer than MAXframe packets are outstanding. The receiving packet controller 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 packet controller and accepted from a receiving packet controller.

A packet controller 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 packet controller 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 packet controller 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 packet controller receiving data from the link clears out its buffers, it sends an RR to the transmitting packet controller. In order to guard against the possibility of the RR being lost and the link becoming permanently locked, the transmitting packet controller will periodically re-transmit the packet that provoked the RNR. The receiving packet controller will continue to respond with RNR until it can accept the packet.

PACKET CONTROLLER COMMANDS

The following conventions are used in this appendix:

Command	Argument(s)	Default:
---------	-------------	----------

Parameter(s)

Packet controller message to operator or user

Operator or user input

Caution or warning

The command name is printed with some letters underlined and in upper case. These letters form the minimum abbreviation that you may use and still have your packet controller understand the command.

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.

Parameters

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

For commands that set values, your packet controller 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.

An asterisk (*) after the default value means the default may vary between models.

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 (hex), numbers by preceding the number with a \$ symbol. When

PACKET CONTROLLER COMMANDS

Parameters

the packet controller shows some of these numeric parameters (those which set special characters), they will be given in hex. The numbers 10 through 15 are denoted by the hexadecimal digits A through F.

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. They may 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 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.

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. For example:

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 B 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.

Command Groups

**Async [A]
Commands**

8bitconv
 AUtoIf
 AWlen
 BBSmsgs
 BReak
 DISPlay
 Echo
 EEscape
 Flow
 LCok
 NUcr
 NULf
 NULLs
 PARity
 RXblock
 Screenln
 TRFlow
 TXFlow
 Xflow

**Character [C]
Commands**

BKondel
 CANline
 CANPac
 COMmand
 DElete
 LCStream
 PASs
 REDisplay
 SENDpac
 STArt
 STOp
 STREAMCa
 STREAMDb
 STReamsw
 Xoff
 XON

**Health and Misc
[H] Commands
and Counters**

BAud
 CALibra
 HEALed
 PORT
 RESET
 RESTART
 SALE
 SOftdcd
 ASYFrerr
 ASYQover
 ASYrxovr
 BBfailed
 DIGISent
 HOvrerr
 HUndrerr
 PMSComm
 RCVDFrmr
 RCVDIfra
 RCVDRrej
 RCVDRnr
 RCVDSabm
 RXAbort
 RXCount
 RXErrors
 RXLenerr
 RXResync
 SENTFrmr
 SENTIfra
 SENTRej
 SENTRNR
 TXCount
 TXQovflw
 TXTmo

**BBS/PMS [B]
Commands**

3rdparty
 AUTOFwd
 CLKSet
 EDithdr
 Forward
 FNPms
 FPms
 Kill
 KILONFWD
 List
 LOGonmsg
 MIne
 MSGHdr
 PMs
 Read
 REMSysop
 SEND (SP, ST)
 SR

**ID [I]
Commands**

Beacon
 BText
 CBell
 CMSg
 MSGDisc
 CText
 CWid
 CWidtext
 CWLen
 HIid
 HOMebbs
 Id
 MYAlias
 MYcall
 MYPcall
 NODetext
 STExt
 Unproto

PACKET CONTROLLER COMMANDS**Command Groups****Link [L]
Commands**

Ax25L2v2
 CONMode
 Connect
 CONOk
 CONPerm
 CONVerse
 CR
 CStatus
 DIGipeat
 DISconnect
 FIrnrnr
 FULLdup
 KISs
 LFadd
 LFIgnore
 MAXframe
 NEwmode
 NOmode
 Paclen
 PASSAll
 REConnect
 REtry
 STATus
 TRACe
 Transparent
 TRies
 TXUifram
 USers
 XMitok

**Printer [P]
Commands**

PRAUtoIf
 PPRint
 PRint
 PRPage
 PRPDisc
 PRPGIn
 PRTTOFf
 PRTTON
 PRTTgl
 PRTUifra

**Monitor [M]
Commands**

ADdrdisp
 AMonth
 BUdlist
 CONStamp
 CRAfter
 DAYUsa
 HEaderIn
 LCALLs
 MAll
 MCOM
 MCon
 MFilter
 MHClear
 MHeard
 Monitor
 MRpt
 MStamp
 PIdcheck

**PMS Sysop
Commands**

3rdparty
 EDithdr
 FNPms
 FOrward
 FPms
 KIll
 KILONFWD
 List
 MIne
 MSGHdr
 PMs
 Read
 SEND
 SP
 SR
 ST

**Timing [T]
Commands**

AXDelay
 AXHang
 CALSet
 CHeck
 CLKadj
 CMdtime
 CPactime
 DAYtime
 DWait
 FRack
 PACTime
 PERSist
 PPersist
 RESptime
 SLOttime
 TXdelay

**PMS User
Commands**

Bye
 Help
 Jlog
 Kill
 KM
 List
 Mine
 Read
 Send
 SR
 Version
 ?

Command List

3rdparty **ON|OFF or Y|N** **Default: ON***

ON - Allows messages to any callsign.
 OFF - Allows messages to be addressed only to/ from MYPcall.
 * Default may vary

Selects whether PMS messages are accepted to other than MYPcall (sysop). When 3rdparty is ON, over-the-air messages may be addressed to any valid callsign. When 3rdparty is OFF, callsigns which do not exactly match MYPcall will be rejected with the message:

No third party mail allowed

The sysop may address messages to any callsign.

8bitconv **ON|OFF or Y|N** **Default: OFF**

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.

ADddisp **ON|OFF or Y|N** **Default: ON**

ON - Addresses on monitored frames will be displayed.
 OFF - Addresses on monitored frames will not be displayed.

The ADddisp command enables or disables the display of the callsign addresses at the beginning of each monitored frame. The command applies only to the terminal display, not printer output. If ADddisp is ON, the following display format will be used for monitored frames:

N4UQQ > N4UQR, TPA5* <UI R> :This is a monitored frame.

If ADddisp is OFF, the same frame will appear as follows:

This is a monitored frame.

PACKET CONTROLLER COMMANDS

Command List

AMonth ON|OFF or Y|N Default: OFF

ON - The month in all system dates will be alphabetic.
 OFF - The month in all system dates will be numeric.

This command allows selection of alphabetic (Jan, Feb,) or numeric (01,02) format of the month in date displays. If AMonth is ON, the following date display format is used:

```
Apr-15-90 19:48:01 (DAYUSA ON) or
15-Apr-90 19:48:01 (DAYUSA OFF)
```

If AMonth is OFF, the following date display format is used:

```
04/15/90 19:48:01 (DAYUSA ON) or
15/04/90 19:48:01 (DAYUSA OFF)
```

AMonth does not affect the PMS date format.

AUTOFwd ON|OFF or Y|N Default: OFF

ON - PMS messages are automatically marked for forwarding.
 OFF - PMS messages are not marked for forwarding.

This command enables automatic *marking* for forwarding of all messages with a callsign in the @ BBS field. This command does not cause forwarding to take place, only marks the message for forwarding. To initiate forwarding, use the FPMS or FNPMS command or wait for your HOME BBS to reverse forward.

AUtoLf ON|OFF or Y|N Default: ON

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.

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: 8 *

n = 7-8, the number of data bits per word *Default may vary

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

PACKET CONTROLLER COMMANDS **Command List**

AX25LV2 **ON|OFF or Y|N** **Default: ON**

ON - The packet controller uses AX.25 Level 2 Version 2.0
OFF - The packet controller uses AX.25 Level 2 Version 1.0

Some implementations of the earlier version of AX.25 protocol won't properly digipeat version 2.0 AX.25 packets. This command exists to provide compatibility with these other packet controllers. See also CHECK.

AXDelay **n** **Default: 0**

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

AXDelay specifies a period of time the packet controller 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 when a standard "voice" repeater is used to extend range. You will have to experiment to find the best value for n. Note that this command acts in conjunction with AXHang.

AXHang **n** **Default: 0**

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

This value can be used to increase channel efficiency when a voice 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 packet controller has heard a packet sent within the hang period, it will not add the repeater keyup delay (AXDelay) to the keyup time.

BAud

NOTE: This command ~~does not apply to all models.~~

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. Of course the RADIO baud rate should not be changed if a connection exists.

The initial portion of the BAud display lists the current data rate of both the terminal port and the radio port 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 selection command prompt.

If the BAud command was entered in error or if you desire just to display the current baud rates, simply enter the letter X <CR> to return to the command prompt without making any changes.

PACKET CONTROLLER COMMANDS

Command List

(BAUD Command Continued)

The display appears as follows:

Radio: 300
Terminal: 1200

Radio	Baud Rates	Terminal
A-	300	L- 300
B-	400	M- 1200
C-	600	N- 2400
D-	1200	O- 4800
E-	1800	P- 9600
F-	2400	Q- 19200
G-	4800	
H-	9600	
I-	19200	
X	Make Changes	

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 correction or additional selections. When you are ready to accept the changes, enter the letter X <CR>. The desired baud rates will take effect immediately after the X response.

BBSmsgs **ON|OFF or Y|N** **Default: OFF**

This command controls how the packet controller displays certain messages in COMMAND and CONVERSE modes.

MESSAGE	EFFECT OF BBSMSGs - ON
*** CONNECTED TO XXXXXX	Message appears on a new line
*** DISCONNECTED	Message appears on a new line
*** retry limit exceeded	Message appears on a new line
*** XXXXXX Busy	Message appears on a new line
*** FRMR sent	Message appears on a new line
*** FRMR received	Message appears on a new line
*** Connect request: XXXXXX	This message is omitted

The BBSmsgs command is primarily useful for host operation with bulletin board software that require link state messages to begin in the first display column.

Beacon **Every|After n** **Default: E 0**

EVERY - Beacon at regular intervals.
AFTER - Beacon after specified interval with no activity.
n = 0 - 250, beacon timing in 10 second intervals.
A value of 0 disables the beacon.

PACKET CONTROLLER COMMANDS

Command List

(Beacon Command Continued)

This command enables beacon sending and causes the first beacon frame to be transmitted. A beacon frame consists of the text of BText in a packet addressed to "BEACON" and sent via the digipeat addresses specified by the Unproto command.

If the keyword Every is specified, a beacon packet is sent every n*10 seconds. 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.

NOTE: If BText is blank, no beacons will be sent.

BKondel **ON|OFF or Y|N** **Default: ON**

ON - <BACKSPACE> <SPACE> <BACKSPACE> is echoed when a char is deleted.
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 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. 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.

BReak **ON|OFF** **Default: ON**

ON - The BREAK signal operates as described in the manual.
OFF - The BREAK signal is ignored by the packet controller.

The BReak command controls enabling or disabling packet controller response to the BREAK signal. The BREAK is a 'space' condition on the serial data input line for longer than two character times which can be generated by many computers and terminals. Some devices will generate an inadvertent BREAK signal during initialization which may force the packet controller into Command mode. Setting BReak to OFF eliminates this.

BText **text** **Default: blank**

text - Any combination of 120 characters and spaces.

BText specifies the content of the data portion of a beacon packet. The default text is blank, i.e., no message. 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.

PACKET CONTROLLER COMMANDS Command List

(BText Command Continued)

If you enter a text string longer than 120 characters, an error message will appear and the command will be ignored.

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

BUdlist **ON|OFF or Y|N** **Default: OFF**

ON - Ignore frames from stations which are not in the LCALLs list.
OFF - Ignore frames from stations which are listed in LCALLs.

Monitor 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.

BYe

A command used by a PMS over-the-air user.

BYE disconnects the user from the PMS with these messages:

```
- Logged off
***DISCONNECTED
```

There will be a short (up to 10 second) delay after the "***logged off" message is received before the PMS issues the disconnect request. If a connection exists to PMS and there is no activity on the radio port for approximately 5 minutes, the PMS will issue a disconnect request.

CALibrate

This command does not apply to all models of PacComm packet controllers. Refer to the TECHNICAL REFERENCE MANUAL for details about calibration procedures and the operation of this command.

CALibrate is an immediate command, and is used to transfer control to the modem tone generation routine. The CALibrate command may be performed at any time without altering the current link state.

The commands available in the calibration routine are:

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

PACKET CONTROLLER COMMANDS **Command List**

CALSet	n	Default: none
---------------	----------	----------------------

n = 0 - 65535 setting the count setting for the calibration routine.

This command does not apply to all models of PacComm packet controllers. Refer to the TECHNICAL REFERENCE MANUAL for details.

The number to be specified by CALSet is determined by the frequency to be calibrated as follows:

modulator tones: $n = (525,000 / f) + 1$
demodulator tones: $n = (262,500 / f) + 1$
Round n to the nearest integer.

CANline	n	Default: \$18 {CTL-X}
----------------	----------	------------------------------

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 other input editing features, is disabled in Trans Mode.

CANPac	n	Default: \$19 {CTL-Y}
---------------	----------	------------------------------

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 CONTROLLER COMMANDS

Command List

(CANPac Command Continued)

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 packet controller. Typing the cancel-output character a second time restores 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 packet controller to commands. Now type another <CTRL-Y>, and type DISPLAY again to show the display is back to normal.

CBell	ON OFF or Y N	Default: OFF
--------------	----------------------	---------------------

ON - A bell character is sent with the connect message.
OFF - No bell character is included in the connect message.

The CBell 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.

CCheck	n	Default: 30
---------------	----------	--------------------

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 packet controller and another station are connected, and the other station “disappears,” the packet controller will try to clean up the link if the specified time elapses without any packets being heard from the other packet controller. This feature depends on the setting of AX25L2V2.

If AX25L2V2 is ON, the packet controller will send a “check packet” if no packets have been heard from it for n*10 seconds. If the receiving packet controller is still connected, it sends an appropriate response packet. If the packet controller initiating the inquiry does not hear a response after RETRY + 1 attempts, it commences a disconnect sequence, as if the Disconnect command had been given.

If AX25L2V2 is OFF and the other station has not been heard for n*10 seconds, the packet controller will not attempt an inquiry, but will send a disconnect frame.

PACKET CONTROLLER COMMANDS **Command List**

CLKADJ **n** **Default: 0**

n = 0 - 65535, specifying the real-time clock speedup factor.

This command allows the clock speed to be increased to compensate for the tendency to lose time when the radio port is very active. If the value of CLKADJ is non-zero, then the correction factor is calculated as: relative clock speed in % = 100 + (9.16667 * 1/n). NOTE: A small value of n has a large effect on the clock speed, a large value of n has a small effect. A value of "0" means no correction will be applied.

CLKSet **ON|OFF or Y|N** **Default: ON**

ON - The controller will accept clock update from home BBS.
OFF - The controller ignores clock updates.

If CLKSet is ON and the HOMEbbs supports clock updating, the date and time of the packet controller will be updated to the BBS date and time when the home BBS connects to the PMS callsign.

CMdtime **n** **Default: 1**

n = 0 - 250, Transparent Mode timeout value in seconds.

This command sets the Transparent Mode timeout value. The same Command Mode entry character is used for exit from both Converse Mode and Transparent Mode. 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 packet controller will exit Transparent Mode and enter Command Mode. You should then see the prompt "cmd:" If n is zero, the only exit from Transparent mode is sending a BREAK to the controller.

CMSg **ON|OFF or Y|N** **Default: OFF**

ON - A text message is sent when a connection is established.
OFF - The text message is not sent.

CMSG enables automatic sending of the message set by CTEXT whenever your packet controller accepts a connect request.

CMSGDisc **ON|OFF or Y|N** **Default: OFF**

ON - The packet controller will disconnect immediately after acknowledgement of the CTEXT sent by CMSG command.
OFF - The controller will not disconnect after sending CMSG.

CMSGDisc controls whether the controller will disconnect after the packet containing connect text (CTEXT) is acknowledged.

If you receive connects with immediate disconnects and never receive any data, it's possible CMSGDISC is (inadvertently) on.

PACKET CONTROLLER COMMANDS

Command List

COMmand *n* **Default:** \$03 {CTL-C}

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 happens.

See CMdtime for information on how the Command Mode entry character is used to escape from Transparent Mode.

CONMode *Conv|Tran* **Default:** *Conv*

Conv - Sets entry to Converse Mode when connection begins.
Tran - Sets entry to Transparent Mode when connection begins.

CONMode controls which mode the packet controller will be placed in after a connection. The connection may result either from a radio connect request or a keyboard command. If you initiate a connection from the keyboard, the timing of the mode change is determined by NEwmode.

If the packet controller 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 to prevent the last part of your command from inadvertently being sent as data.

Connect *call#1 [Yia call#2[,call#3...,call#9]]*

call#1 - Callsign of packet controller to be connected to.
call#2 - Optional callsign of station to act as digipeater.

Connect is an immediate command. It initiates a connect request to packet controller *call#1*, optionally through one or more digipeaters.

The bracketed part of the command line, "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, *call#1*.

If NEwmode is ON, the packet controller will immediately enter Converse or Transparent Mode, as specified by the CONMode. If NEwmode is OFF, the packet controller will enter Converse or Transparent Mode when the connection is successfully completed.

PACKET CONTROLLER COMMANDS

Command List

(Connect Command Continued)

An error message is returned if the controller is in a connected state, or is attempting to connect or disconnect. If no response to the connect request occurs after REtry attempts, the command is aborted and a failure message is displayed.

CONOk **ON|OFF or Y|N** **Default: ON**

ON - Connect requests will be accepted.
OFF - Connect requests will be rejected.

This command determines the action taken by the packet controller when it receives a connect request. 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 packet controller is not in Transparent Mode, a connect request will display this message.

connect request: XXXXXX(callsign of station attempting to connect)

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

*** XXXXXX station busy (callsign to which connect was attempted)

CONPerm **ON|OFF or Y|N** **Default: OFF**

ON - The current connection on the current stream will not be allowed to enter the disconnected state.
OFF - The current stream connects and disconnects normally.

CONPerm ON forces the packet controller 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.

CONStamp **ON|OFF or Y|N** **Default: OFF**

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.

PACKET CONTROLLER COMMANDS

Command List

(CONStamp Command Continued)

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]
```

CONVerse

NOTE: The letter **K** is an alias for CONVerse allowing convenient single key transfer to converse mode.

CONVerse is an immediate command which causes the packet controller to change from Command Mode into Converse Mode. Any link connections are not affected. Once in Converse mode, everything you type is packetized and transmitted. Typing the Command Mode entry character returns the packet controller to the Command Mode.

CPactime **ON|OFF or Y|N** **Default: OFF**

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.

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 periodic packetizing, 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., <LF> <CR>), at which point the packet controller will treat <CR> as an ordinary character.

CPactime ON provides a mode of operation in which your text is transmitted soon after you type it, but in short bursts of a few characters. It eliminates the delays while long packets are being typed.

CR **ON|OFF or Y|N** **Default: ON**

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.

(CR Command Continued)

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 packet controller, not as data to be included in the packet, and 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, set LFadd ON, or the other station can set AUtoLF ON.

CRAfter **ON | OFF** **Default: ON**

ON - A carriage return <CR> is sent after each frame.
 OFF - Monitored frames are displayed without adding a <CR>.

A carriage return character <CR> (\$0D) is normally added to the end of each monitored frame. CRAfter allows the elimination of the <CR> character for proper display of some types of frames. CRAfter affects both the terminal and printer ports.

CStatus

CStatus is an immediate command which shows the stream identifier and link state of all streams (links), the current input and output streams, and whether or not each stream is "permanent" (see CONPerm). An example of a CStatus display:

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

The example above shows the A stream is the currently selected or active stream for both input and output, 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.

CText **text** **Default: blank**

text - Any combination of up to 120 characters and spaces.

CTEXT specifies the text of the packet to be sent after a connection is made, if CMSG is ON. The default text is blank, 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 CTRL-V)

PACKET CONTROLLER COMMANDS

Command List

(CText Command Continued)

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, use a RESET command, or enter % or &.

CWid **Every | After** **n** **Default: E 0**

EVERY- Send CWID at regular intervals.
AFTER- Send CWID once if any activity during interval.
n = 0 - 250 specifying CWID timing in 100 second intervals.
A value of 0 disables the CWID.

The CWid command enables CW identification (if CWidtext not blank) and sets the timing interval. Every/After works similarly to Beacon command except n is the number of 100 millisecond intervals. (Export only)

CWidtext **text** **Default: blank**

text- Any combination of 32 characters, numbers, and spaces which have significance in International Morse.

Up to 32 characters of text to be sent as CWID. Accepts all alphabetic, numeric, and common punctuation characters. The length of CWID is limited by the packet controller PTT watchdog timer. At 20 wpm, about 13 characters may be sent. (Export only)

CWlen **n** **Default: 6**

n - 1 to 7 specifying the relative length of CWID dots.

The CWID speed is set by specifying the relative dot length. The default CWlen setting of 6 yields approximately 20 wpm. A CWlen of 7 yields 10 wpm, and a CWlen of 3 yields 40 wpm. (Export only)

DAYtime **yymmddhhmm** **No Default**

yymmddmmss - Current date and time to set.

This command allows you to set the current date and time for the packet controller. The format 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 date & time parameter is used by the commands CONStamp and MStamp to "time stamp" receivedframes. Entries in the "heard log" (displayed by MHeard) are time stamped if date&time has been set. If you type DAYtime without a parameter, the controller will display the current date and time.

PACKET CONTROLLER COMMANDS **Command List**

DAYUsa **ON|OFF or Y|N** **Default: ON***

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

This command determines the format for the packet controller'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.

DELete **ON|OFF or Y|N** **Default: OFF**

ON - The delete char editing key is <DELETE> (\$7F).
OFF - The delete char 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 packet controller indicates the deletion is controlled by the BKondel command.

You cannot 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 REDisplay command.

DIGipeat **ON|OFF or Y|N** **Default: ON**

ON - The packet controller will digipeat packets.
OFF - The packet controller will not digipeat packets.

When this parameter is ON, any packet received that has your packet controller'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. Digipeating takes place concurrently with other packet controller operations and does not interfere with normal packet operation.

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

Disconnect Command Continued)

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 packet controller moves to the disconnected state. If a disconnect command is entered while the packet controller 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 packet controller is in Transparent Mode.

DISPlay [class]

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

Async	display asynchronous port parameters
Bbs	display PMS parameters
Character	display special characters
Health	display health counters and LED status
Id	display ID parameters
Link	display link parameters
Monitor	display monitor parameters
Printer	display printer 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.

```
cmd:DISP < CR >
```

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

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

This value is used to avoid collisions with digipeated packets. The packet controller 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. If all stations specify a proper wait time the digipeater will capture the frequency every time it has data to send, since digipeated packets are sent without this delay.

PACKET CONTROLLER COMMANDS **Command List**

Echo **ON|OFF or Y|N** **Default: ON**

ON - Characters received from the computer or terminal are echoed by the packet controller.
OFF - Characters are not echoed.

This command controls local echoing by the packet controller 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 everything you type, set Echo OFF.

EDithdr **n to-call[-n] @-call[-n]**

n message number to be edited.
to-callsign revised callsign to be placed in 'TO:' location.
@-callsign revised callsign to be placed in '@ BBS:' location.
n optional SSID.

The EDithdr command allows the editing of PMS message headers to allow readdressing of messages (changing the TO: callsign) and revising the callsign of the '@ BBS' field. The EDithdr command will accept either two or three parameters; the message number and either or both callsigns are required. It does not allow editing the BID. If the parameters are not correct, the following message will be displayed:

Parameters are missing

To revise only the 'TO:' callsign of message 9, enter the following:

cmd:EDithdr 9 W4MUA <CR>

The PMS will respond with the following display:

Msg #	Stat	Date	Time	To	From	@ BBS	Subject
4	T	07/15/90	09:54	W1AW	WS4Z	@ W4DPH	VE tests
New message header							
Msg #	Stat	Date	Time	To	From	@ BBS	Subject
4	T	07/15/90	09:54	W4MUA	WS4Z	@ W4DPH	VE tests

Similarly, to revise only the '@ BBS:' callsign of message 6, enter:

cmd:EDithdr 6 @W4DPH <CR>

To revise both callsigns in message 3, enter the following:

cmd:EDithdr 3 N4UQQ @ K0ZXF <CR>

To remove the callsign in the '@ BBS' field in message 2 and leave the field blank, enter the following:

cmd:EDithdr 2 @ <CR>

The PMS will respond with the 'before' and 'after' displays in each case.

PACKET CONTROLLER COMMANDS Command List

EScape **ON|OFF or Y|N** **Default: OFF**

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

Some terminals interpret the <ESCAPE> character as a special command prefix. If you have such a terminal, you can protect yourself from unexpected text sequences by setting EScape ON. See also the MFilter command.

Firmnr **ON|OFF or Y|N** **Default: OFF**

ON - Firmnr is active.
 OFF - Firmnr is disabled.

This command is used to prevent the frequent polling of remote TNCs when they are busy. Works around the AX.25 protocol specification for busy-poll timing. This command operates properly with remote TNCs having FIRMRNR set to ON and with most network nodes.

Flow **ON|OFF or Y|N** **Default: ON**

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 (4) 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.

FOrward **n**

n = the number of the message to be *marked* for forwarding.

This command allows messages to be *marked* for forwarding. The PMS does not support automatic forwarding to other BBS (to reduce channel congestion), but supports manual forwarding (see FPms, FNPms commands) and automatic reverse forwarding. If you have an arrangement with your local BBS Sysop to forward BBS mail to your PMS, then the PMS will reverse forward marked messages to the local BBS when the BBS connects to deliver your mail. If you type:

PACKET CONTROLLER COMMANDS

Command List

(FORward Command Continued)

```
cmd:FORward n <CR>
```

The PMS will respond with:

```
Message marked for forwarding
```

If you try to mark a bulletin for forwarding, the PMS will display:

```
You cannot forward a bulletin
```

The FORward command also is used to remove the forwarding flag from a marked message. To undo the forwarding of a message, enter:

```
cmd:FORward n <CR>
```

The PMS will respond with the message:

```
Message forwarding reset
```

FNPms call#1 [Via call#2]

```
call#1 -Callsign of node or switch to be connected to.  
call#2 -Optional callsign of station to be digipeated through.
```

This command causes the PMS to connect to a node or switch (call#1) via no more than one digipeater (call#2).

Once this connection exists, the PMS sends the node the connect string contained in NODetext to cause the node to connect to the desired BBS/PMS station. Once that connection is established, the PMS uploads all messages marked for forwarding.

If the station connected to by the NODetext is the HOMEbbs, the standard BBS forwarding format will be used for transferring messages, otherwise the PMS will follow the standard "human" format.

If the FNPms command is used to connect to a station which is not a node, the NODetext will be sent but will be interpreted as data and no further action will take place. After approximately five minutes, the PMS will time-out and disconnect.

If the station connected to by the NODetext command string is not a BBS or PMS capable of receiving PMS messages, or if a forwarding error occurs, automatic disconnection takes place.

FPms call#1 [Via call#2]

```
call#1 -Callsign of BBS/PMS station to be connected to.  
call#2 -Optional callsign of station to be digipeated through.
```

This command causes the PMS to connect to a BBS/PMS station (call#1) via no more than one digipeater (call#2). The callsign string may not exceed 23 characters. Once this connection exists, the PMS uploads all messages marked for forwarding.

PACKET CONTROLLER COMMANDS

Command List

(FPms Command Continued)

This command may be used to upload messages to another PMS or human operator as well as a conventional BBS. If the station connected to by the FPms command is the HOMEbbs, the standard BBS forwarding format will be used for transferring messages, otherwise the PMS will follow the standard "human" format.

If the FPms command is used to connect to a station which is not a BBS or PMS capable of receiving PMS messages, or if a forwarding protocol error occurs, automatic disconnection occurs.

FRack *n* **Default: 3 [HF 6]**

n = 1-15, marking frame acknowledgment timeout in seconds.

After transmitting a packet requiring acknowledgment, the packet controller waits the frame acknowledgment timeout before incrementing the retry counter and sending the frame again. If the packet address includes digipeaters, the Retry interval is adjusted to $n * (2 * m + 1)$ where *m* is the number of 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 packet controllers repeatedly send packets which collide.

FULLdup **ON|OFF** or **Y|N** **Default: OFF**

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

When full duplex mode is off, the packet controller 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 packet controller ignores the DCD signal and acknowledges packets individually. It should not be used unless both your station and the station you are communicating with are full-duplex stations.

HEaderIn **ON|OFF** or **Y|N** **Default: OFF**

ON - The header/text of monitored packets use two lines.
OFF - The header/text of monitored packets use one line.

This command affects the display format. If HEaderIn is OFF, the address information is displayed with the packet:

KV7D > N2WX: Go ahead and transfer the file.

If HEaderIn 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.

PACKET CONTROLLER COMMANDS Command List

HEAlled **ON|OFF or Y|N** **Default: OFF**

ON - The controller will "dither" the CON and STA LEDs.
OFF - The CON and STA LEDs operate normally.

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

When HEAlled is set ON, the two LEDs flash in an alternate fashion. The LEDs will not flash if the software is not operating.

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

H(elp or ?)

A command used by a PMS over-the-air user. Displays a list of PMS commands with a brief description of each.

B(ye)	B [CR] disconnects you from PMS
H(elp)	H [CR] or ? [CR] displays this help file
J(log)	J [CR] displays a list of callsigns heard with date/time
K(ill)	K n [CR] deletes message n (only to/from your callsign)
KM	Kill all messages addressed to users callsign
L(ist)	L [CR] lists the 10 latest messages
M(ine)	M [CR] lists the 10 latest messages to/from your callsign
R(ead)	R n [CR] reads message number n
S(end)	S (callsign) (@ BBS callsign) [CR] begins a message addressed to (callsign) at home BBS of (@ BSBcallsign) Subject: max 28 characters ending with [CR]. Text: End each line with [CR]. End message by typing /ex[CR] or CTRL-Z [CR] at the beginning of a new line.
SP	Send message marked Personal
ST	Send message in NTS format
V(ersion)	V [CR] displays the software version of the PMS system.

HIId **ON|OFF or Y|N** **Default: OFF**

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 packet controller. If HIId is OFF, the packet controller will never send an identification packet. If HIId is ON, the packet controller will send an identification packet every 9.5 minutes if the station is digipeating packets. The ID command allows the operator to send a manual identification packet.

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

HOMebbs **callsign-n**

Callsign - Up to six alphanumeric characters.
 n - Optional SSID from 0 to 15.

This command sets the callsign of the 'home BBS' which forwards mail to your PMS or which you normally access manually.

Id

Id is an immediate command which sends 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. See **HIId** for ID frame information.

LDlog

A command used by a PMS over-the-air user.

Displays a log of the callsigns most recently heard by the PMS host packet controller. The number of entries in the display may vary based on PMS buffer space. The display is in the following format:

N4UQQ*	PMS	07/15/90	18:33
TPA5	NODE	07/15/90	18:32
K0ZXF	BBS	00/00/00	00:00

Zeros are shown in the date and time fields if the clock was not set. An asterisk (*) means the station was heard via a digipeater.

(K)ll **n**

A command used by a PMS over-the-air user. See **KIll** below.

(KM)

A command used by a PMS over-the-air user. Kills all messages addressed to the user's callsign. See **KIll** below.

KEll **n**

n - The number of the message you wish to kill.

This command allows deletion of a message in the PMS. If the message is found and erased, the PMS will display:

Message erased

If an invalid message number is given, the PMS will display:

Message not found

KIll O(ld) will delete the 10 oldest messages.

PACKET CONTROLLER COMMANDS **Command List**

KILOWFD **ON|OFF or Y|N** **Default: ON**

ON - Messages are deleted after forwarding.
OFF - Messages are retained after forwarding.

This command selects whether messages which have been marked for forwarding will be deleted or retained after forwarding takes place.

KISS **ON|OFF or Y|N** **Default: OFF**

ON - **RESTART** command initiates KISS firmware.
OFF - **RESTART** command reinitiates standard packet firmware.

This command supports the use of TCP/IP, MICROSAT software, and other software which specifies the KISS mode.

The KISS command sets the KISS parameter. Then the **RESTART** command will cause the controller to initialize using the KISS mode. The KISS setting is preserved in bbRAM.

The KISS mode may be exited by removing the battery jumper (which destroys bbRAM parameters), by entering the command **PARAM AX0 255** at the **TCP/IP NET >** prompt or from another communication program. To exit KISS when using an MS-DOS computer and communication software: 1. Hold down the ALT key, 2. Press keys 1 9 2 on the numeric keypad, 3. Release the ALT key, 4. Hold down the ALT key again, 5. Press keys 2 5 5 on the numeric keypad, and 6. Release the ALT key.

LCalls **call#1,call#2...,call#8** **Default: blank**

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 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 the only ones you want to listen to.

If you want to monitor selected callsigns, enter those calls in **LCalls** and set **BUDlist ON**. If you want to ignore selected callsigns, enter the calls to ignore in **LCalls** and set **BUDlist OFF**.

LCok **ON|OFF or Y|N** **Default: ON**

ON - The controller will send lower case chars to the terminal.
OFF - Lower case characters are changed 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

PACKET CONTROLLER COMMANDS

Command List

(LCok Command Continued)

disabled in Transparent Mode. Input characters and echoes are not translated. Since echoes of the characters typed in are not translated to upper case, if both operators set LCok OFF, each operator can type messages in lower case and see incoming packets displayed in upper case.

LCStream **ON|OFF or Y|N** **Default: ON**

ON - The packet controller will convert the character following the STReamsw character to upper case before processing it.
OFF - The packet controller will process the character immediately following the STReamsw character as it is entered.

When operating multi-connect, the user must enter a stream identifier (default A through J) after the STReamsw(itch) 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 or Y|N** **Default: OFF**

ON - A Linefeed <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. If the receiving station reports overprinting of packets from your station, set LFadd ON. This command is disabled in Transparent Mode.

LFIgnore **ON|OFF or Y|N** **Default: OFF**

ON - Line Feed characters <LF> are ignored.
OFF - Line Feed characters <LF> are acted upon.

This command controls whether packet controller responds to ASCII Line Feed (\$0A) characters in COMMAND and CONVERSE modes. When ON, line feeds are totally ignored. 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 the LFadd command description. The transmitting station may have the LFadd command set ON inadvertently.

List

A command used by a PMS over-the-air user. See List below.

List

This command lists all messages in the PMS. A typical display:

Msg #	Stat	Date	Time	To	From	@ BBS	Subject
4	T	07/15/90	09:54	W1AW	WS4Z	@ W4DPH	VE tests
3	P	07/16/90	09:23	W1BEL	K0ZXF	@ W4DPH	Hello
2	PF	07/15/90	23:45	K0ZXF	W1BEL	@ K0ZXF	Bulletins
1	B	07/13/90	03:52	ALL	N4UQQ	@ W4DPH	Help

15003 Bytes free
Next message Number 5
cmd:

The values for Status are B(ulletin), P(riate or Personal), and (NTS) T(raffic). Messages to 'ALL' will have a P(riate) status but will be readable by all users. The PMS will accept and display Bulletins, but cannot originate bulletins.

LOGonmsg ON|OFF or Y|N Default: ON

- ON - The standard PMS log-on message is sent.
- OFF - The standard PMS log-on message is not sent.

This command controls sending of the standard PMS log-on message. If LOGonmsg is ON, the following message is sent:

```
*** CONNECTED TO (PMS callsign)
Logged on to (PMS callsign)'s Personal Message System
Optional PMS system operator supplied message (STExt)

CMD(B/H/J/K/KM/L/M/R/S/SR/V/?) >
```

If LOGonmsg is OFF, the sign-on appears as follows:

```
*** CONNECTED TO (PMS callsign)
Optional PMS system operator supplied message (STExt)

CMD(B/H/J/K/KM/L/M/R/S/SR/V/?) >
```

MAIL ON|OFF or Y|N Default: ON

- ON - Monitors both "connected" and "unconnected" packets.
- OFF - Monitors only "unconnected" packets.

This command determines the class of packets which are monitored. If MAIL is OFF, only eligible packets (determined by BUdlist and LCALLs commands) sent by other packet controllers in the unconnected mode are displayed.

If MAIL is ON, all otherwise eligible frames are displayed, including those sent between two other connected packet controllers.

PACKET CONTROLLER COMMANDS **Command List**

MAXframe **n** **Default: 4**

n = 1 - 7, signifying a number of frames.

MAXframe sets a limit on the number of unacknowledged frames which the packet controller may have outstanding at any time. This is also the maximum number of contiguous packets which can be sent during any transmission.

For file transfers, and other 'high volume' data applications, there are optimum values for MAXframe and PACLen which may be different than the parameters set for interactive communication. The values also differ based on the quality of the RF link.

MCOM **ON|OFF or Y|N** **Default: OFF**

ON - All frames are monitored.
OFF - Only information frames are monitored.

MCOM causes the display of control frames when Monitor is ON. When MCOM is OFF, only 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. See the protocol chapter for more information.

As with other monitor commands, the stations monitored are determined by BUDlist and LCALLs.

MCon **ON|OFF or Y|N** **Default: OFF**

ON - Monitor mode remains active when in connected state.
OFF - Monitor mode is off when in connected state.

If MCON is ON, the Monitor command will enable monitoring while your packet controller is in a connected state. If MCON is OFF, the display of monitored packets is suspended when a connect occurs, and is resumed when disconnected.

MFilter **n1 [n2 [n3 [n4]]]** **Default: 00 (null)**

n = 0 - \$7F, specifying an ASCII character code to be filtered.
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 you want to filter the <CTRL-L> screen clearing character, you can set MFilter 12. If you also want to eliminate <CTRL-Z> characters, which some computers interpret as

PACKET CONTROLLER COMMANDS

Command List

(MFilter Command Continued)

end-of-file markers, you can set MFILTER 12,26. Clear the MF list by the command MFilter 00. Do not enter 0 or \$00 as the first entry of the MFilter string as any characters entered following that entry will be ignored.

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. Clear the list of stations heard when you first begin to monitor the packet activity.

MHeard

MHEARD is an immediate command. It causes the packet controller 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. The maximum number of heard stations that can be logged is 18. As more stations are heard, earlier entries are dropped. Logging of stations is disabled when PASSALL is ON.

If the date and time are set, entries in the heard log will be time stamped. Stations which can be identified by their transmissions as either a PMS, BBS, or NODE are so listed in the heard log.

MIne

A command used by a PMS over-the-air user. See MIne below.

MIne

Lists all messages to or from your PMS callsign (MYPCALL). For example, if MYPcall is set to W1BEL, MIne will show the following messages from the example shown above in the LIST command:

Msg #	Stat	Date	Time	To	From	@ BBS	Subject
3	P	04/16/90	09:23	W1BEL	K0ZXF	@ W4DPH	Hello
2	PF	04/15/90	23:45	K0ZXF	W1BEL	@ K0ZXF	Bulletins

15003 Bytes free
Next message Number 4
cmd:

Monitor ON|OFF For Y|N Default: ON

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

If Monitor is ON and the packet controller is not in Transparent Mode, packets not addressed to your packet controller will be

PACKET CONTROLLER COMMANDS

Command List

(Monitor Command Continued)

displayed. The addresses in the packet are displayed along with the data portion of the packet. The calls are separated by a comma and the sub-station ID field (SSID) is displayed if it is other than zero.

The MAIL, BUdlist, and LCAlls commands determine which packets are to be monitored. The MCon command controls the action of monitor mode when the packet controller is connected. All monitor functions are disabled in Transparent Mode.

The format of the monitor display is controlled by HHeaderIn. See also MRpt and MStamp.

MRpt **ON|OFF or Y|N** **Default: ON**

ON - Display digipeating stations for monitored packets.
OFF - Display only 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.

```
WB9FLW > AD7I:Hi Paul.
```

If MRpt is ON, the entire path is displayed, and stations that have already relayed the packet are indicated with an asterisk.

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

MSGHdr **ON| OFF or Y|N** **Default: ON**

ON - Forwarded messages will have a message header inserted.
OFF - Forwarded messages will not have a message header.

This command selects whether forwarded messages will have a message header inserted at the beginning of the message.

The message header appears as follows:

```
Originated From: < callsign > mm/dd/yy hh:mm Msg # n
```

The format of the date depends on the setting of DAYUsá.

MStamp **ON|OFF or Y|N** **Default: OFF**

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 are set initially by the DAYtime command, and the date format is determined by the DAYUsa command.

PACKET CONTROLLER COMMANDS **Command List**

MYAlias	call[-n]	Default: blank
call - Alternate identity of your controller. 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.		
This command permits HId to identify normally with the MYcall-specified callsign yet permit an alternate (alias) repeat-only "callsign."		
MYcall	call[-n]	Default: NOCALL-0
call - Callsign of your packet controller. n = 0 - 15, an optionally specified sub-station ID (SSID).		
This command tells the packet controller what its callsign is. This callsign will be placed in the FROM address field for all packets it originates. 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 call. The SSID will be 0 unless explicitly set to another value.		
MYPcall	call[-n]	
Call - Up to six alphanumeric characters. n = 0 - 15, an optionally specified sub-station ID (SSID).		
This command sets the callsign of the PMS, which must not be the same as MYcall . Normally use MYcall with a different SSID.		
NEwmode	ON OFF or Y N	Default: OFF
ON - Entry to data transfer mode occurs when the Connect command is given, and return to command mode is automatic at disconnection.		
OFF - Switching to data transfer mode occurs a connection but no return to command mode occurs at disconnection.		
The NEwmode command may be used to select the way the controller behaves when connections are made and broken.		
If NEwmode is OFF, the packet controller will remain in Command Mode after you issue a Connect command until a connection is actually established, then will enter Converse Mode or Transparent Mode, depending on the setting of CONMode. When the connection is terminated, the packet controller remains in Converse or Transparent Mode until you force it to return to Command Mode.		

PACKET CONTROLLER COMMANDS

Command List

(NEWmode Command Continued)

If NEWmode is ON, the packet controller will enter the specified conmode as soon as you issue a Connect command, without waiting for the connection to be established. Anything typed thereafter will be packetized to be transmitted once the connection is complete. When the connection is broken, or if the connect attempt fails, the packet controller returns to Command Mode.

NODetext text **Default: blank**

text - A valid connect string (limited to one digipeater).

The NODetext field should be filled with a valid connect string or left blank. (See the Connect command for examples of a proper connect string and also note the limitations below).

This connect string is used by the FNPms command to cause the node or switch in use to connect to the desired station for message forwarding. This may be the HOMEbbs or another station for forwarding of PMS messages. The NODetext should contain the C(onnect) command followed by the callsign of the station to be forwarded to. Only one digipeater field may be included and must be preceded by the V(ia) command.

NOMode ON|OFF or Y|N **Default: OFF**

ON - The controller will only switch modes (command, converse or transparent) upon explicit command.
OFF - The packet controller switches modes in accordance with the setting of NEWmode.

When NOMode is ON, the packet controller 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 or Y|N **Default: OFF**

ON - <NULL> chars are sent to the terminal after <CR>.
OFF - <NULL> characters are not sent after <CR>.

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 to allow time for the carriage to return and the paper to advance. Set NUcr ON if your terminal misses characters after a <CR> .

PACKET CONTROLLER COMMANDS Command List

NULf	ON OFF or Y N	Default: OFF
<p>ON - <NULL> characters are sent to the terminal after <LF> characters.</p> <p>OFF - <NULL> characters are not sent after <LF> chars.</p> <p>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 NULs. This delay is required by some display terminals. Set NULf ON if your terminal sometimes misses characters at the beginning of the line.</p>		
NULs	n	Default: 0
<p>n = 0 - 30, the number of nuls added after <CR> or <LF>.</p> <p>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 nuls are to be sent after <CR> , <LF> , or both.</p> <p>Devices requiring nuls after <CR> are typically hard-copy devices requiring time for carriage movement. Devices requiring nuls after <LF> are typically CRTs which scroll slowly. Extra null characters are sent only in Converse and Command Modes.</p>		
Paclen	n	Default: 128 *
<p>n = 0 - 255, specifying the maximum length of the data portion of a packet. The value 0 is equivalent to 256.</p> <p>* NOTE: The default Paclen value may vary according to model.</p> <p>The packet controller 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.</p>		
PACtime	Every After n	Default: After 10
<p>Every - Packet timeout occurs every n*100 milliseconds</p> <p>After - Packet timeout occurs when n*100 milliseconds elapse with no input from the computer or terminal.</p> <p>n = 0 - 250, specifying 100 ms intervals.</p> <p>This parameter is always used in Transparent Mode, and will 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. If n = 0, packets are generated with no wait time.</p>		

PARity	n	Default: 0 (none) *
<p>n = 0 - 3, selecting a parity option according to the following code: 0 = none, 1 = odd, 2 = none, 3 = even.</p>		
<p>* NOTE: Parity default value may vary on some models.</p>		
<p>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.</p>		
PASs	n	Default: \$16 {CTL-V}
<p>n = 0 - \$7F, specifying an ASCII character code.</p>		
<p>This command selects the ASCII character used for the "pass" input editing command. The parameter n is the ASCII code (hex or decimal) for the character typed in order to include the following character in a packet or text string.</p>		
<p>You can use this character to send any character in packets, even though that character may have some special function for the packet controller. A common use for the pass character is to allow <CR> to be included in the BText and CText messages. You can include <CR> in text when you are in Converse Mode, to send multi-line packets. (The default send-packet character is <CR>.)</p>		
PASSAll	ON OFF or Y N	Default: OFF
<p>ON - Controller will accept packets with invalid CRCs. OFF - Controller will only accept packets with valid CRCs.</p>		
<p>This command causes the packet controller to display packets received with invalid CRC fields if they consist of an even multiple of 8 bits and up to 339 bytes.</p>		
<p>The packet controller will attempt to decode the address field and display the callsign(s) in the standard monitor format, followed by the text of the packet.</p>		
<p>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.</p>		
<p>Logging of stations heard (for display by MHeard) is disabled whenever PASSAll is ON, since the callsigns detected may be incorrect.</p>		

PACKET CONTROLLER COMMANDS

Command List

PErsist	n	Default: 127
<p>n = 0 to 255 setting the probability of retransmitting a frame.</p> <p>PErsist is a threshold setting which is applied to a random number between 0 and 255 generated when a frame needs to be resent. If the random value is greater than the PErsist value, no retry is made. If the random value is less than or equal to PErsist, a retry attempt is made. The default value of 127 equates to a 50% probability that channel access will be attempted. PErsist = 255 provides 100% probability of attempting to seize the channel.</p>		
PIdcheck	ON OFF or Y N	Default: OFF
<p>ON - All frames with a PID other than \$F0 are ignored. OFF - All frames are accepted equally.</p> <p>When Pidcheck is ON, all frames with Protocol Identifier (PID) other than \$F0 are ignored to reduce the screen clutter from network nodes. Pidcheck OFF ignores the PID.</p>		
PMs	ON OFF or Y N	Default: OFF
<p>ON - The Personal Message System is enabled. OFF - The Personal Message System is disabled.</p> <p>This command enables or disables PMS operation via the radio port. When PMS is off, a connect request to MYPcall is sent the messages:</p> <div style="border: 1px solid black; padding: 5px;"><p>*** (PMS callsign) BUSY *** DISCONNECTED</p></div> <p>Turning the PMS off will not destroy the messages in the PMS memory area. Use the Kill command to remove unwanted messages. To reset the messages to begin with number 1, the RESET command must be used, which will also destroy all other battery-backed parameters and PMS messages.</p>		
POrt	n	Default: 2
<p>n = 1 or 2, specifying a radio port (1 = HF, 2 = VHF).</p> <p>NOTE: This command does not apply to all models.</p> <p>The POrt command is used to select the radio port (on models which feature switchable ports) and engages the proper set of parameters for the selected port. If the POrt command is given without a parameter, the controller responds with the currently selected port. If a command is given to switch ports while a connection exists, the controller responds with the message:</p> <div style="border: 1px solid black; padding: 5px;"><p>? not while connected</p></div>		

PACKET CONTROLLER COMMANDS

Command List

PPersist	ON OFF or Y N	Default: OFF
ON - Selects P Persistence method of retry timing. OFF - Selects FRACK (conventional) method of retry timing. PErsist, PPersist, and SLOttime implement the P Persistence method of timing retried frames. This superior method of retry timing will not be effective unless all stations on a channel use it. Consult with other packet operators in your local area to encourage use of PPersistence.		
PPRint	n - n	
n - PMS message number to be sent to printer port. n - n - Range of PMS message numbers to be sent to printer port. NOTE: This command does not apply to all models. This command allows the printing of Personal Message System (PMS) messages to an attached printer.		
PRAUtoLf	ON OFF	Default: ON
ON - A linefeed <LF> character is sent following each carriage return <CR> sent to the printer. OFF - Carriage return characters are sent to the printer without an added linefeed character. NOTE: This command does not apply to all models. This command functions like the AUtoLf command, but applies to the printer port. If your printer requires a linefeed character with each carriage return, set PRAUtoLf to ON. If printed material appears to be double spaced and it should be single spaced, set PRAUtoLf to OFF.		
PRint	ON OFF	Default: OFF
ON - The printer port is enabled. OFF - The printer port is disabled. NOTE: This command does not apply to all models. This command allows the printer port functions to be enabled or disabled. If PRint is OFF, no printing will occur under any condition. If PRint is ON, printing will be controlled by the link connection state and PRITgl state.		
PRPage	ON OFF	Default: ON
ON - The printer output will automatically be paginated. OFF - The printer output will be non-paginated. NOTE: This command does not apply to all models.		

PRPage Command Continued

This command is in conjunction with the PRPGln command to paginate printed output.

If PRPage is ON, a formfeed character will be sent to the printer after PRPGln number of lines have been printed.

If the printer automatically paginates printed data, or if no pagination is desired, set PRPage OFF.

PRPDisc **ON|OFF** **Default: ON**

ON - A formfeed character will be sent to the printer when the link state becomes disconnected.

OFF - No formfeed character will be sent to the printer.

NOTE: This command does not apply to all models.

This command allows the printer to begin printing at the top of page position each time a connection is established.

PRPGln **n** **Default: 60**

n - number of lines printed between formfeed characters.

NOTE: This command does not apply to all models.

This command sets the number of lines to be printed on each page. The PRPGln value is used by PRPage to paginate output.

PRTOFF **n** **Default: \$0F**

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

NOTE: This command does not apply to all models.

If PRTTgl is ON, printer output may be controlled by the commands PRITON and PRITOFF imbedded in the Converse mode data which is received.

The PRITOFF command sets the value of the character in received data which is used to toggle printer output off.

PRITON **n** **Default: \$10**

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

NOTE: This command does not apply to all models.

If PRTTgl is ON, printer output may be controlled by the commands PRITON and PRITOFF imbedded in the Converse mode data which is received.

The PRITON command sets the value of the character in received data which is used to toggle printer output on.

PACKET CONTROLLER COMMANDS

Command List

PRTTgl **ON|OFF** **Default: ON**

ON - Printer output in converse mode will be controlled by PRTTON and PRTOFF characters in received
OFF - Printer output occurs for all data in CONVERSE mode.

NOTE: This command does not apply to all models.

The PRTTgl command controls whether printer output may be controlled by commands imbedded in the Converse mode data which is received.

PRTUifra **ON|OFF** **Default: OFF**

ON - Monitored UI (unproto) frames are printed.
OFF - Only connected frames are printed.

NOTE: This command does not apply to all models.

The PRTUifra command allows printing of UI frames on the printer. It is not intended for monitoring of connected transmissions in the same sense that the Monitor command applies to terminal output. There is no mechanism to detect duplicate received frames, therefore these duplicate frames will be printed. This command should be used in conjunction with the LCALLS and BUdlist commands to limit data received to only one station.

Frames are printed without callsign address headers, to allow the 'reconstruction' of text or graphics documents sent in the UI mode. The command CRAfter controls whether a carriage return character is appended to the end of each monitored frame.

(Read)

A command used by a PMS over-the-air user. See Read below.

Read **n - n**

n - The number of the PMS message you wish to read

n - n - The first and last numbers of the messages to be read

This command displays message number n in the format below:

```
Stat      : P
Posted   : 07/15/90 19:33
To       : K4NTA
From     : W1BEL
@ BBS    : K0ZXF
BID      :
Subject  : FADCA Meeting
```

Text of message.

cmd:

PACKET CONTROLLER COMMANDS

Command List

(Read Command Continued)

If the command Read is given without a number, a listing of messages in the PMS will be displayed (same as List command). Reading any PMS message (not necessarily a newly entered message) will reset the blinking PMS or STA LED.

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

call#1 - Callsign to be reconnected to.
call#2 - Optional callsign(s) to be digpeated 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 packet controller 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 of the parameter list, see the Connect command.

REDisplay n Default: \$12 {CTL-R}

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

This command is used to change the redisplay-line input editing character. The parameter n (hex or decimal) is the ASCII code for the character you type in order to redisplay the current line.

When you type the redisplay-line character, type-in flow control is temporarily released (if it was enabled), displaying 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 is shown. You are now ready to continue typing.

Use the redisplay-line character to see a "clean" copy of your input with a printing terminal after deleting characters. If you have set BKondel OFF, deletions are designated with <BACKSLASH> characters, rather than by trying to correct the input line display.

You may 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.

REMSysop ON|OFF Default: OFF

ON - The MYPcall is allowed sysop priveleges over the air.
OFF - The MYPcall is not allowed sysop priveleges over the air.

When REMSysop is ON, the callsign contained in MYPcall is allowed to read and kill any message from over the air. This allows the sysop to remotely manage the message base in the PMS equipped packet controller.

RESET

This is an immediate command. It resets all parameters to default settings and re-initializes the packet controller.

WARNING: Parameters, heard lists and PMS messages are lost.

To re-initialize the packet controller using battery-backed RAM parameter values, use **RESTART** or turn the controller off and on.

RESPtime **n** **Default: 0**

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 may be used to increase throughput during operations such as file transfer when the sending packet controller usually sends the maximum number of full-length packets. Occasionally, the sending packet controller 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 packet controller sets **RESPtime** to 10, these collisions will be avoided.

RESTART

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

RETry **n** **Default: 10**

n = 0 - 15, specifying the number of retransmissions.

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. A value of 0 for **n** specifies an infinite number of retries. If the number of retries is exceeded, the packet controller goes to the disconnected state (with an informative message if not in **Transparent Mode**). See also the **FRack** and **PPersist** commands.

RXblock **ON|OFF or Y|N** **Default: OFF**

ON - The controller sends data to the terminal in **RXblock** format
OFF - The controller sends data to the terminal in **normal** format.

PACKET CONTROLLER COMMANDS

Command List

(RXblock Command Continued)

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 packet controller-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 received data unit header.

When RXblock is on, data from other stations will be sent from the packet controller in the following format:

	\$FF	L0	L1	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, parameters like AUtolf, MFilter etc. should be set OFF to prevent uncertainties in the size of the data field.

SAle

This is an immediate command which displays to the screen whatever service information the dealer has inserted in the EPROM. If no data has been stored in the EPROM, SAle returns the message:

not implemented

Screenln n Default: 0

n = 0 - 255, specifying the character width of the terminal platen or screen.

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 leave SCREENLN = 0 to avoid conflict.

(S)end

A command used by a PMS over-the-air user. See SEND below.

SEND To-call-n @ call-n \$ BID

To-call - Callsign to which you wish the message addressed.
 @-call - The 'home BBS' callsign of the station to which the message is sent (if known)
 n - Optional SSID
 \$ BID - Optional Bulletin Identifier

This command begins the procedure for creating a message addressed to (To-call). Addressing messages to 'ALL' is permitted. If the operator does not enter a callsign, the message will be addressed to MYPCALL, the PMS callsign.

The home BBS callsign (if known) should be included as shown above. The @ symbol must be typed between the callsigns; it is not provided automatically.

The BID (Bulletin Identifier) is entered beginning with a US Dollarsign (\$). The remainder of the BID field should be filled according to local practice.

You will be prompted for the subject and text.

Subject: A maximum of 28 characters ending with a <CR>. If too long a subject is entered, the PMS will truncate the part exceeding 28 characters.

Text: End each line with a <CR>. End the message by typing . (period) <CR>, /ex, or CTRL-Z at the beginning of a new line. Message size is limited only by the unused memory space available in the PMS. If that space is exceeded, the PMS will display:

- Buffer overflow

When the message has been saved, the PMS will display:

Message saved as Msg # n

SEndpac n Default: \$0D <CR>

n = 0 - \$7F, specifying an ASCII character.

This command selects the ASCII code (parameter n) for the character that will force input to be packetized and queued for transmission in Converse Mode. You can enter the code in either hex or decimal.

For conversation, set SEndpac = \$0D and CR ON. This causes packets to be sent at natural intervals, and causes the <CR> to be included in the packet. If CPactime is ON, set SEndpac to some value not ordinarily used (say, <CTRL-A>), and set CR OFF. This allows forcing packets to be sent, but will not result in extra <CR> characters being transmitted in the text.

PACKET CONTROLLER COMMANDS

Command List

Slottime **n** **Default: 1**

n = 0 to 255, specifying SLOTTIME in 10 ms increments.

Slottime selects the length of the delay between generating random numbers as part of the PERSIST retry timing. Larger SLOTTIME values lower the packet controller's priority in accessing the channel.

Softdcd **ON | OFF** **Default: See Below**

ON - The DCD signal is derived from firmware.
OFF - The DCD signal is obtained from the modem hardware.

This parameter should only be set to ON when using the TNC-220 which is not equipped with a tuning indicator. **SOFTDCD** MUST be set to OFF for proper operation of all other models.

On packet controllers which use the 8530 controller IC, the DCD signal may alternatively be obtained from either the 8530 flag detection feature or from the conventional modem DCD signal.

SP

This is an extension of the PMS SEND command. SP causes the message status to be marked with a P designator for personal or private. Refer to the SEND command.

SR **n**

n = Number of PMS message which is to be replied to.

SR stands for Send Reply. The message number must be included; the SR command does not send a reply to the most recently read message. If no message number is included, the message is displayed:

parameters are missing

SR automatically reverses the TO and FROM callsigns from message number **n** and enters the same subject in the reply message. No entry is made in the @BBS field since this information may not be reliably determined from all received messages. Use the EDIthdr command to edit in the @BBS callsign if one is needed. This command is especially useful for replying to messages directly entered into the PMS since no @BBS entry is needed.

```
cmd:l
Msg # Stat Date   Time To      From    @ BBS   Subject
  11 BR 06/28/90 10:04 N4UQR  W4DPH
  12 BR 06/28/90 12:49 W4DPH  N4UQR
11206 Bytes free
```

ST

This is an extension of the PMS SEND command. ST causes the message status to be marked with a T designator for National Traffic System (NTS) traffic. The text of such a message should be filled with an NTS formatted message. Refer to the SEND command.

STArt **n** **Default: \$11 {CTL-Q}**

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

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.

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 packet controller is disabled, and the packet controller will only respond to hardware flow control (CTS). If the same character is used for both the User Restart and User Stop characters the packet controller will alternately start and stop transmission upon receipt of the character.

STATUS

\$00 - Disconnected state.
\$01 - Connect is in progress.
\$03 - Disconnect in progress.
\$04 - A connection exists and all frames are acknowledged.
\$06 - A connection exists and unacknowledged frames exist.

STATUS is an immediate command which returns a hexadecimal value indicating selected link parameters on the current stream.

STExt **text** **Default: blank**

text - a string of up to 80 text chars ending with carriage return.

STExt sets the optional text which is sent as the PMS custom sign-on message. If STExt is blank, no message is sent. LOGonmsg controls the standard sign-on message and is not related to STExt.

STOp **n** **Default: \$13 {CTL-S}**

n = 0 - \$7F, specifying an ASCII character.
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 packet controller to the terminal. This is the character used to halt the packet controller's terminal output and stop scrolling.

PACKET CONTROLLER COMMANDS

Command List

(STOp Command Continued)

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 packet controller is disabled, and the controller will only respond to hardware flow control (CTS).

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

STREAMCa ON|OFF or Y|N Default: OFF

ON - Callsign of other station is displayed.

OFF - Callsign of other station is not displayed.

This command is used to enable the display of the connected-to station callsign after the stream identifier. It is somewhat analogous to the use of MRpt to show digipeat paths when monitoring.

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

<i>STREAMCa ON</i>	<i>STREAMCa OFF</i>
A :K4NTA:hi howie	A hi howie
hello ted how goes it?	hello ted how goes it?
B :N7CL:*** CONNECTED to N7CL	B *** CONNECTED to N7CL
A Ted, AZ on other stream	A Ted, AZ on other stream

Thus, what looked like “|B” now appears as “|B: <callsign>:”. This option is very useful for human operators trying to operate multiple simultaneous connections.

STREAMDb1 ON|OFF or Y|N Default: OFF

ON - Double all received STReamswitch characters.

OFF - Do not “double” received STReamswitch characters.

This command is used to display received streamswitch characters by “doubling” them. The following example illustrates this action. With STREAMDb1 ON, and STReamsw(itch) set to “|”, the following might be displayed from your packet controller:

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

In this case the sending station actually transmitted:

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

The same frame received with STREAMDb1 OFF would appear as:

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

PACKET CONTROLLER COMMANDS Command List

(STREAMDbI Command Continued)

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 packet controller.

NOTE: The STReamsw(itch) character must NOT be one of the letters A-J.

STReamsw *n* **Default:** \$7C <|>

n = 0 - \$FF, specifying an ASCII character.

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

The character can be PASsed in CONVerse 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).

Also see the STREAMDbI and STREAMCall commands.

TRACe ON|OFF or Y|N **Default:** OFF

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, provide trace information if possible.

A trace display will appear in four columns on an 80-column display. Following is an example of a compressed trace display.

The frame shown in the example would be monitored as follows:

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

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

```
000: 86A24040 40406096    CQ 0KV7B 0KF    @@'.n.@@'.  
010: 6E844040 E103F074    7B Px:449.49.0 n.@@ this is a
```

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

TRACe Command Continued)

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 sub-station ID byte and the protocol control bytes.

Transparent

This is an immediate command. It causes the packer controller 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.

TRFlow **ON|OFF or Y|N** **Default: OFF**

ON - Transparent Mode terminal flow control can be enabled.
OFF - Transparent Mode software flow control is disabled.

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 packet controller 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 packet controller 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 packet controller to control output from the terminal.

TRies **n** **Default: NONE**

n = 0-15, specifying the RETRY level on the current 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 packet controller has an outstanding unacknowledged frame, it will return the current number of tries; if the packet controller has no outstanding unacknowledged frames, it will return the number of tries which were 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.

PACKET CONTROLLER COMMANDS

Command List

(REtry Command Continued)

This command is useful for obtaining statistics on the performance of a given path or channel.

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

TXDelay **n** **Default: 30**

n = 0 - 120, specifying 10 ms intervals.

This value tells the packet controller 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.

TXFlow **ON|OFF or Y|N** **Default: OFF**

ON - Transparent Mode software flow control is enabled.
OFF - Transparent Mode software flow control is disabled.

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 packet controller will use only hardware flow control and all data sent to the terminal remains fully transparent.

If TXFlow and Xflow are ON, the packet controller 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 packet controller.

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.

TXUifram **ON|OFF or Y|N** **Default: ON**

ON - UI (unproto) frames may be transmitted.
OFF - No UI (unproto) frames will be transmitted.

This command eliminates dumping of unsent frames as UI frames when a connection fails. It is useful to operators of Packet Bulletin Board systems. TXUifram ON will also prevent sending UI frames from the keyboard in Converse mode, but will not interfere with the sending of beacons.

PACKET CONTROLLER COMMANDS **Command List**

Unproto **call#1** [**Y**ia **call#2** [, **call#3** .., **call#9**]] **Default: CQ**

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**

n = 1 - 10, the number of connections allowed

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 packet controller. 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, etc.

(Y)ersion

A command used by a PMS over-the-air user. See Version below.

Version

The Version command causes the entire packet controller firmware sign-on banner to be displayed, including the features supported, firmware release date, and EPROM checksum.

Xflow **ON|OFF or Y|N** **Default: ON**

ON - XON/XOFF (software) flow control is enabled.
OFF - XON/XOFF flow control is disabled.

If Xflow is ON, the computer or terminal is assumed to respond to the packet controller Start and Stop characters set by XON and XOFF. If Xflow is OFF, the packet controller will communicate flow control commands via RTS.

XMitok **ON|OFF or Y|N** **Default: ON**

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

PACKET CONTROLLER COMMANDS**Command List**

(XMITOK Command Continued)

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

You might use this command to insure that your packet controller 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 {CTL-S}
-------------	----------	------------------------------

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

This command selects the TNC Stop character (hex or decimal), which is sent by the packet controller to the computer to stop input from that device.

This character would normally be set to <CTRL-S> for computer data transfers. If you want notification in Converse mode when the buffers are filled, set this character to <CTRL-G> (\$07), which rings the terminal bell.

XON	n	Default: \$11 {CTL-Q}
------------	----------	------------------------------

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

This command selects the TNC Restart character (hex or decimal), which is sent by the packet controller to the computer to restart input from that device.

This character would ordinarily be set to <CTRL-Q> for computer data transfers. If you want notification in Converse mode when the buffers are filled, set this character to <CTRL-G> (\$07), which rings the terminal bell.

MESSAGES**General Messages**

|A

PacComm [model] packet controller
 AX.25 Level 2 Version 2.0
 FEATURES:
 Release X n.n.n Y mm/dd/yy - nnK RAM
 Checksum \$xx
 cmd:

This sign-on message that appears when you turn on your packet controller or issue the RESET command. The release number indicates the software version. The checksum is a hex number for comparison with the checksum given in the firmware documentation. FEATURES: is followed by a list of firmware features in the EPROM.

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 packet controller to load the default parameters from ROM. (This will be the case the first time you turn on your packet controller) This message also appears if the packet controller loads the defaults in response to the RESET command.

cmd:

This is the Command Mode prompt. When this prompt appears, the packet controller is waiting for you to issue a command. Anything you type after this prompt will be interpreted as a command to the packet controller. 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 REDisplay) to retype the prompt.

was

Whenever you change the setting of one of the packet controller parameters, the previous value will be displayed. This confirms that the packet controller properly interpreted your command, and reminds you of what you have done.

too many packets outstanding

This message would appear in response to a CONVerse or Transparent 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 packet controller, an error message will be printed. You may see any of the following messages depending on the type of error you have made.

?parameter

The command was typed correctly, but the remainder of the line couldn't be interpreted.

?call

The callsign argument that does not meet the packet controller 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.

?unknown command

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

?not enough

Not enough arguments for a command that expects several parameters.

?not while connected

An attempt to change MYcall, AX25L2V2 or POrt was made while in a connected or connecting state.

MESSAGES

Link Status Messages

?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.

?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 packet controller 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 packet controller is not in the disconnected state, the packet controller 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 packet controller is connected to and the digipeater route if any. The callsign sequence is the same sequence you would enter to initiate the connection.

*** LINK OUT 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 Disconnect command, the connect 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 Disconnect command, the packet controller will go immediately to the disconnected state.

Link state is: FRMR in progress

The packet controller is connected but a protocol error has occurred. An improper implementation of the AX.25 protocol could cause this state to be entered. The packet controller 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 packet controller will inform you whenever the link status changes. The link status may change in response to a command (Connect or Disconnect), a connect or disconnect request 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 packet controller 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 packet controller 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 packet controller types this message it also sends a DM packet (busy signal) to the station that initiated the connect request. If the packet controller rejects a connect request because you have set

MESSAGES

Link Status Messages

CONOk OFF, you can issue your own request to the station that called.

*** DISCONNECTED

This message is displayed whenever the packet controller goes to the disconnected state from any other link state. This message may be preceded by a message explaining the reason for the disconnect.

*** 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.

*** callsgn busy
*** DISCONNECTED

This message indicates that your connect request was rejected by a DM packet (busy signal) from the other station. A PacComm packet controller 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 packet controller is connected, and a protocol error has occurred. The packet controller 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 packet controller 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 packet controller is in Transparent Mode.

PMS Messages

This section describes messages generated by the Personal Message System (PMS). The term **USER** is applied to an over-the-air PMS user, and **SYSOP** to the packet operator. Messages to the user are 'SENT', while messages to the Sysop are 'DISPLAYED.'

PMS Response Messages

[PMS-3.0-C\$]

SID transmitted by PMS during forwarding sessions to the HOMEbbs.

*** CONNECTED TO (PMS callsign)
 Logged on to (PMS callsign)'s Personal Message System
 (optional message line copied from SText - may be blank)

CMD(B/H/J/K/KM/L/M/R/S/SR/V/?) >

This message set is sent when a user connects to the PMS. Either one of the above message lines may or may not be displayed based on PMS parameter settings.

*** PMS-callsign busy
 *** DISCONNECTED

If the Sysop is entering a message from the computer terminal, or is PMS is OFF, a user attempting to connect over the air will receive a 'busy' message.

PMS IS BUSY

This message is sent if the PMS is in use by an over-the-air user, and the system operator attempts to enter or kill a message or any other action which would alter the contents of the PMS memory area.

-Logged off
 *** DISCONNECTED

These messages are sent in response to the Bye command from a PMS user.

MYPCALL's Personal Message System

Is displayed to the connecting station if the PMS station has the parameter LOGON set to ON.

You have mail waiting.

Displayed to an over-the-air user during sign on to PMS if there is mail addressed TO that callsign.

MESSAGES

PMS Messages

B(ye)	B [CR] disconnects you from PMS.
H(elp)	H [CR] or ? [CR] displays this help file.
J(log)	J [CR] displays a list of callsigns heard (optional date/time)
K(ill)	K n [CR] deletes message n (only to/from your callsign).
KM(ine)	KM [CR] deletes all READ messages addressed to your call.
L(ist)	L [CR] lists the 10 latest messages.
M(ine)	M [CR] lists the 10 latest messages to/from your callsign
R(ead)	R n [CR] reads message number n.
S(end)	S (callsign) [CR] begins a message addressed to (callsign). Subject: max 28 characters ending with [CR]. Text: End each line with [CR]. End message by typing /ex [CR] or CTRL-Z [CR] at the beginning of a new line.
SR(eply)	SR n [CR] Sends a reply to message n prompting only for text.
V(ersion)	V [CR] displays the software version of the PMS system.

The help message display is sent in response to the Help or ? command from a PMS user.

Message erased

This message is sent or displayed in response to the Kill command which originates from either the Sysop or user.

Message marked for forwarding

Response to the Sysop command to mark a message for reverse forwarding.

Message forwarding reset

Response to the Sysop command to 'unforward' a message previously marked for forwarding.

Message has been sent to the printer queue

Response to the PPRint Sysop command to print a PMS message on the packet controller printer port.

Message saved as Msg # n

Response sent or displayed after the completion of 'Sending' a message by either the Sysop or user.

nnnnn Bytes free, Next message Number n

Displayed at the end of the LIST display

PMS Error Messages**[PMS CMD ERR]**

This message is issued when a PMS user sends a command in an improper format. The most common error is using more than one character of the command name. Sysop command errors receive the normal ?unknown command response.

Message not found

This message is issued whenever a Read or Kill command is given with an invalid message number by either a user or Sysop.

YOU CANNOT KILL THIS MESSAGE

This message is issued when a user attempts to kill a message not to or from his callsign.

YOU CANNOT READ THIS MESSAGE

This message is issued when an attempt is made to read a message not addressed to or from the user's callsign, or ALL.

No third party mail allowed

When 3rdparty is OFF, this message is issued when a user attempts to send a message to other than the PMS (Sysop) callsign.

- Buffer overflow

This message is issued when the PMS memory capacity is exhausted and the message being sent has been truncated.

Parameters are missing

This message is issued when the EDithdr command is used by the Sysop with improper parameters.

You cannot forward a bulletin

This message is issued when an attempt is made by the Sysop to mark a bulletin for forwarding.

Entry too long, try again

Displayed whenever too many characters are entered at a prompt i.e. too long a BID or Subject entry.

Index

< BACKSLASH >	14	BYE.....	38,55
< BACKSPACE >	5,14,54,86	CALIBRA.....	55
< RR R F > type of display	41,75	CALSET.....	56
A cknowledgements.....	7,10,26	CANLINE.....	13,16,56
Address	40	CANPAC.....	16,18,56
ASC11 codes for control char.....	13	CBELL.....	24,57
Automatic disconnect	11	CHECK.....	27,57
Automatic operations	22	CLKADJ.....	31,58
AX25 control codes	42	CLKSET.....	58
AX25 Level 2.....	40	CMDTIME.....	19,58
bbRAM	16	CMSG.....	23,58,62
BBS operation	32	CMSGDISC.....	23,58
Beaconing	23	COMMAND.....	13,18,59
Beginning operation	6	CONMODE.....	16,59
Break	8,20,54	CONNECT.....	7,9,59,60
Busy signal	23,60	CONOK.....	23,60
Callsign.....	6,9,36,78	CONPERM.....	60
Carriage return.....	3	CONSTAMP.....	31,60-1,63-4
Clock	31	CONVERSE.....	18,59,61,93
cmd:.....	4	CPACTIME.....	61,80,89
Command alphabetical listing	50	CR.....	24,61,89
Command format.....	37	CRAFTER.....	12,62
Command Groups	48	CSTATUS.....	34,62
COMMAND mode	46	CTEXT.....	23,58,62-3
Command mode entry.....	8,19	CWID.....	63
Command parameters.....	46	CWIDTEXT.....	63
Commands:		CWLEN.....	63
8BITCONV	18	DAYTIME.....	31,63,77
ADDRDISP	12,50	DAYUSA.....	31,64,77
AMONTH	31,51	DELETE.....	13,16,18,64
AUTOFWD.....	51	DIGIPEAT.....	9,64
AUTOLF	5,16,51,87	DISCONNECT.....	8,64-5
AWLEN	4,51,81,88	DISPLAY.....	11,28,65
AX25L2V2.....	27,52,57	DWAIT.....	26,32,65,87
AXDELAY.....	25,43,52	ECHO.....	5,16,66
AXHANG.....	25,52	EDITHDR.....	36,37,66
BAUD	6,52-3	ESCAPE.....	19,67
BBSMSGs	32,53	FIRMNR.....	67
BEACON.....	23,53-4,96	FLOW.....	16,19,21,67
BKONDEL.....	14,16,18,54,64,86	FNPMS.....	37-8,68
BREAK.....	8,13,20,54	FPMS.....	37-8,68-9
BTEXT	23,54-5,81	FRACK.....	25,69
BUDLIST.....	28,55,74-5,96	FULLDUP.....	31,69

HEADERLN.....	11,69,76	PPERSIST.....	26,83
HEALED.....	28,70	PPRINT.....	37,83
HELP.....	38,70	PRAUTOLF.....	83
HID.....	22,64,70,78	PRINT.....	83
HOMEBBS.....	35,71	PRPAGE.....	83
ID.....	22,71,78	PRPDISC.....	84
JLOG.....	38,71	PRPGLN.....	84
KILL.....	37-8,71	PRTTOFF.....	84
KISS.....	15-7,72	PRTTON.....	84
LCALLS.....	27,55,72,77,96	PRITGL.....	84,85
LCOK.....	16,19,72	PRTUIFRA.....	85
LCSTREAM.....	33-4,73	READ.....	35-6,38,85
LFADD.....	51,73	RECONNECT.....	86
LFIGNORE.....	16,19,73	REDISPLAY.....	15,18,86
LIST.....	37-8,73-4	REMSYSOP.....	86-7
LOGONMSG.....	34-5,74	RESET.....	16,87
MALL.....	27,74,77	RESPTIME.....	27,44,87
MAXFRAME.....	42,75	RESTART.....	60,72,87
MCOM.....	42,75	RETRY.....	10,59,87
MCON.....	11,75	RXBLOCK.....	32,87
MFILTER.....	57,66,75-6,87	SCREELN.....	5,16,19,88
MHCLEAR.....	12,76	SEND.....	37,88-9
MHEARD.....	12,76	SENDPAC.....	18,62,89
MINE.....	37,76	SLOTTIME.....	27,89,90
MONITOR.....	11,65,76-7,90	SOFTDCD.....	33,90
MSGHDR.....	35,77	SP.....	37,90
MRPT.....	11,77	SR.....	37-8,90
MSTAMP.....	63,77	ST.....	37,90-1
MYALIAS.....	9,78	START.....	18,91
MYCALL.....	6,9,70,78	STATUS.....	32,91
MYPCALL.....	35,78	STEXT.....	35,37,91
NEWMODE.....	17,59,78	STOP.....	18,91
NOMODE.....	17,79	STREAMCALL.....	32,92
NODETEXT.....	35,68,79	STREAMDBL.....	33,92
NUCR.....	5,16,19,79	STREAMSW.....	33,93
NULF.....	5,16,19,80	TRACE.....	41,93
NULLS.....	5,16,19,80	TRANSPARENT.....	16,19,94
PACLEN.....	24,26,80	TRFLOW.....	21,94
PACTIME.....	19,24,27,61,80	TRIES.....	11,94
PARITY.....	4,81	TXDELAY.....	8,25,52,95
PASS.....	16,18,62,81	TXFLOW.....	21,95
PASSALL.....	25,76,81	TXUIFRAM.....	32,95
PERSIST.....	26-7,82	UNPRONTO.....	70,96
PIDCHECK.....	82	USERS.....	33,96
PMS.....	34-8,82,104	XFLOW.....	21,96
PORT.....	32,82	XMITOK.....	23,96-7

XOFF	18,19,97	Double characters	5,65
XON	16,18,97	Doublespaced lines	6,52,73
Computer & radio interfacing	1	Entering commands	3,15-6
Connections-unsuccessful	10	Entering data transfer modes	16-7
CONTROL.....	40	FCC notice	vi
Control-C.....	8,13	FCS.....	40
Control-R.....	14	File transfer	19
Control-S/Control-Q	15	Filtering characters.....	28
Control-V	15	FLAG.....	40
Control-X.....	14	Flow control	15,20
CONVERSE mode.....	16	Full duplex operation	31-2,69
Counters.....	29-31	Garbled display.....	5
ASYRXOVR.....	29	Group conversations	27
BBFAILED.....	29	HDLC.....	40
DIGISENT	29	Health counters	28
HOVRERR.....	29	Hexadecimal(HEX).....	13
HUNDRERR	29	Identification.....	22,71
RCVDFRMR.....	29	Inhibiting transmitting.....	23,97
RCVDIFRA	29	Input editing.....	3,13
RCVDREJ.....	29	Interference.....	vi
RCVDSABM.....	29	Level.....	2
RXCOUNT	30	Link Status Messages	
RXERRORS.....	30	***callsign busy	102
SENTFRMR	30	***CONNECTED to:.....	100
SENTIFRA.....	30	***connect request:.....	101
SENTREJ	30	***D ISCONNECTED	101
TXCOUNT	30	***LINK OUT OF ORDER	100
TXQOVFLW	31	***retry count exceeded	10,102
TXTMO	31	FRMR sent:string.....	102
CSMA	43	FRMR received:	102
DATA	40	Link state is:	
Date format	31	CONNECT in progress.....	101
DCD	7,33,69	CONNECTED to call#1via	100
Digipeater routing.....	9,54	DISCONNECT in progress.....	101
Digipeating	9,54	DISCONNECTED	102
DISPLAY.....	65	FRMR in progress	101
ASYNCR.....	65	Lost characters	4
BBS.....	65	MCOM display.....	42
CHARACTER.....	65	Messages	
HEALTH.....	65	?already connected to station....	100
ID	65	?parameter	3,99
LINK	65	?call	99
MONITOR.....	65	?clock not set.....	99
PRINTER.....	65	?unknown command	3,99
TIMING.....	65	?not enough.....	99
Display formatting	6,26	?not while connected.....	99
Display width.....	6,88	?not while disconnected.....	99

?range.....	100	RTS/CTS	20
?too long	100	Rubout.....	14
?too many	100	Satellite operation	32
?VIA.....	100	Sending control characters.....	15
bbRAM loaded with defaults	98	Serial port parameters.....	4
cmd:.....	4,98	Serial RS-232/TTL port.....	1
too many packets outstanding	99	Sign-on message	3
was.....	99	Software requirements	2
Monitoring.....	11,27	Special characters	13
Digipeater routing	8,9	Special input characaters	13
Heardlog.....	12	SSID	22
Selective	27	Telephone modem software	2
While connected	11,27	Time stamping.....	31
Multiple connections.....	33	Transfer mode entry.....	78
Operating modes	15	Transmitter timing.....	25
Operating statistics	28	TRANSPARENT mode	16
Packet protocol	39	Exit.....	19
Packet timing.....	25	Timing	27
Parameter loss	15	Typing corrections.....	3,5,13
PID.....	40	Undisplayed characters.....	4
PMS Messages		VIA.....	9,22,59
***CONNECTED TO	103	Voice repeater	25
***(PMS callsign)busy.....	103	XOFF/XON.....	97
[PMS CMD ERR]	105	Warranties.....	ii
[PMS-3.0-C\$].....	103		
- Buffer overflow.....	105		
- Logged off.....	103		
(H)elp	104		
Entry too long, try again.....	105		
Message erased	104		
Message marked-forwarding	104		
Message forwarding reset	104		
Message sent to prntr	104		
Message saved as Msg#n.....	104		
Message not found.....	105		
MYPCAL's PMS	103		
nnnn Bytes,free next Number	104		
No third party mail allowed	105		
Parameters are missing.....	105		
You cannot forward a bulletin ...	105		
YOU CANNOT READ THIS..	105		
YOU CANNOT KILL THIS	104		
You have mail waiting.....	103		
Protocol	27,39		
Radio interfacing	2		
Repair service.....	ii		

SIMPLE C64 TERMINAL PROGRAM

Revised April 15, 1989

```
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  POKE 53280,1: POKE 53281,1: POKE 53272,23
25  REM;USE THIS FOR LINE 20 FOR VIC-20
    : 20 POKE 36879,25
30  OPEN 2,2,3,CHR$(40)+CHR$(96)
40  POKE 36869,242
45  IF A$=CHR$(133) THEN A$="CONTROL C"
50  PRINT CHR$(144)
60  C$=CHR$(147)
70  K$=CHR$(20)
80  J$=CHR$(187)
90  CR$=CHR$(13)
100 PRINT C$;"WAIT.."
110 GOSUB 300
120 PRINT C$;"POWER-UP YOUR TNC NOW"
130 PRINT J$;
140 GET #2,A$
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$="" THEN 140
210 PRINT#2,CHR$(M%(ASC(A$)));
220 GOTO 140
300 DIMI%(255),M%(255)
310 FOR Z=0 TO 64: M%(Z)=Z:NEXT: M%
    (13)=13: M%(20)=8: M%(160)=32
320 FOR Z=65 TO 90: Y=Z+32: M%(Z)=Y: NEXT:
    FOR Z=91 TO 95: M%(Z)=Z: NEXT
330 FOR Z=193 TO 218: Y=Z-128: M%(Z)=Y: NEXT
340 M%(133)=03: M%(134)=19: M%(135)=17: M%
    (136)=16
350 FOR Z=0 TO 255: Y=M%(Z): IF Y<>0 THEN I%
    (Y)=Z
360 NEXT Z
370 RETURN
```

TECHNICAL REFERENCE MANUAL

for

PacComm Packet Controller Models

TINY-2 and MICROPOWER-2

PacComm Packet Radio Systems, Inc.

3652 W. Cypress Street
Tampa, FL 33607-4916 USA
(813) 874-2980

Copyright 1989, 1991 by PacComm Packet Radio Systems, Inc.

ALL RIGHTS RESERVED

Reproduction or translation of any part of this work beyond that permitted by Sections 107 or 108 of the 1976 United States Copyright Act (or its legal successor) without the express permission of the copyright owner is unlawful. Requests for permission to copy or for further information should be addressed to PacComm Packet Radio Systems, Inc.

Second Edition, Second Printing - February 1991

PacComm Warranties

PacComm Packet Radio Systems, Inc. (PacComm) warrants "Assembled and Tested" amateur 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, PacComm 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, used in commercial service or which has been repaired outside PacComm facilities without permission. Consequential damages are not covered.

PacComm provides a rapid repair service for our products, whether in or out of warranty. Shop rates are modest. Carefully pack the unit and send it to the PacComm 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.

TABLE OF CONTENTS

INTRODUCTION	1
Features	1
NET/ROM and ROSE Compatibility	1
COMPUTER INTERFACING	2
RS-232 Port Signals	2
Nonstandard RS-232C Serial Ports	3
Serial Port Operation At TTL Levels	4
Serial Data Signal Definitions	4
RS-232C Handshaking Protocol	6
RADIO INTERFACING	7
Radio Connector (J2) Pin Definitions	7
Direct Connection to Microphone and Speaker	8
Accessory Jack Connection	8
External Interface Box.	8
HARDWARE DESCRIPTION	9
LEDs	9
Detailed Circuit Description	9
Modem Disconnect - J5	11
ADJUSTMENTS	14
Jumper Functions	14
JP B : Lithium Battery	14
JP D: NET/ROM + RFDCD	14
JP L: Digital Loopback	14
JP RA: Audio Input Impedance	14
JP Aux: Auxiliary I/O	14
JP 8: Power Connection	14
Trimpot/Varicap Adjustments	14
Baud Rate Setting	15
Audio Input Level	15
Transmit Level Setting	15
Modem Receive Bias Adjustment	16
Carrier Detect Level	16
Crystal Oscillator Adjustment	16

TABLE OF CONTENTS

TRUBLESHOOTING	17
Loopback testing	18
Analog loopback	18
Digital loopback	19
General Tests	19
Step 1: Obvious Problems	19
Step 2: Power Supply	19
Computer Terminal Interface	19
Packet controller won't sign on to the computer.	20
The controller prints only gibberish.	20
The controller won't accept commands.	20
Other symptoms	21
TINY-2/MICRO-2 appears dead	21
Modem won't key transmitter.	21
Uncopyable transmitted packets	21
Uncopyable received packets	22
DCD LED flickers excessively or stays on	22
Transmitter locks in transmit condition	22
APPENDICES	A
Node Interconnection Cables	A
Dual TINY-2/MICRO-2 Node	A
Dual Node, TINY-2/MICRO-2 and TNC-2(00)	A
Triple Node	A
Computer Interfacing Diagrams	B
Computer Interfacing Diagrams	C
Radio Interfacing Diagrams	D
Radio Interfacing Diagrams	E
RS-232 Signal Table	F
Bit-Shifted ASCII Character Table	G

INTRODUCTION

INTRODUCTION

This manual applies to both the TINY-2 and MICROPOWER-2 Packet Controllers. These units share the same basic design and interface to radios and computer terminals identically. The term TINY-2/MICRO-2 is used throughout this manual to refer to both units when dealing with components and procedures which are the same for both units.

The terms "computer and terminal" are used interchangeably to refer to the computer or terminal used to communicate with the packet controller. This is not meant to imply that a computer is necessary for operation of a packet controller.

Every effort has been made to make this manual complete and accurate, however PacComm assumes no responsibility for damages incurred as a result of using this documentation. PacComm reserves the right to make changes to equipment specifications or documentation at any time without notice or obligation to previous purchasers. Your comments and suggestions for improvement are welcome.

Be sure to read any Errata, manual updates, and software release notes before operating the packet controller.

Features

Firmware

- Completely TNC-2 firmware Compatible
- Supports all 'TAPR' commands plus many PacComm additions.
- Personal Message System (optional on TINY-2)

Hardware

- NMOS Z-80 CPU and SIO (HDLC) [MICROPOWER-2:CMOS]
- 4.9 MHz CPU clock speed
- 32K EPROM and 32K RAM
- Typical Power Consumption - 9-14 vdc @ 220 ma. [40ma]
- Crystal controlled IC modem supports 1200/2200 Hz tones
- DE-9P (male) for RS-232 computer
- TNC-2 type modem disconnect header
- Power, Connect, Status, PTT, Carrier Detect LEDs
- Extruded aluminum case 5 x 7 x 1 3/8 inches with gray, oven baked wrinkle finish
- Metal end panels grounded to prevent RFI
- MICROPOWER-2 printer port and hardware clock options

NET/ROM and ROSE Compatibility

The TINY-2/MICRO-2 are TNC-2 compatible meaning that an EPROM designed for a TNC-2 compatible (such as both NET/ROM and ROSE networking EPROMS) will operate in a TINY-2/MICRO-2. Please refer to Appendix A for interconnection cable information for dual/multi-node units.

COMPUTER INTERFACING

The TINY-2/MICRO-2 packet controller communicates with the computer or terminal through a serial port using either RS-232C or "TTL" level signals. One of these signal levels should properly interface to most computers. See the Appendices to this manual for specific computer interfacing instructions.

RS-232 Port Signals

The serial port connectors on the TINY-2/MICRO-2 packet controller are on the rear panel marked "RS-232" and "TTL Computer." The pins on the serial port connector of packet controller that must be connected are shown in the following table. This is commonly called a "three-wire" RS-232 cable.

RS-232 Serial Port "3 Wire" Cable Wiring for DB-25

Computer DB-25 TINY-2/MICRO-2

1 and 7 -----	5
2 -----	3
3 -----	2

RS-232 Serial Port "3 Wire" Cable Wiring for DE-9

Computer DE-9 TINY-2/MICRO-2

5 and shell -----	5
2 -----	2
3 -----	3

The packet controller is configured as Data Communications Equipment (DCE), the technical term for an RS-232C modem. Most computers and terminals are configured as Data Terminal Equipment (DTE). Specifically, the packet controller connector is an IBM "AT style" using a male DE-9 connector, wired as DCE rather than DTE. If you use an "AT" PC or portable computer with a 9-pin serial port, you can wire cable connector pins 2 through 8 directly (pin 2 to pin 2, etc.)

Pin 1 on the TINY-2/MICRO-2 is the Data Carrier Detect (DCD) signal which goes high (true) when the packet controller is in the "connected" state. This is analogous to the DCD signal on telephone modems, i.e. the pin is asserted when a communication connection exists. Pin 9 is used as an auxiliary radio carrier detect input. This pin may also be used when two or more packet controllers are mated for use with network firmware such as the NET/ROM or ROSE EPROM. More detailed information is presented later in the manual.

Some computers require that Clear to Send (CTS) of the computer serial port connector (pin 5 of a DB-25 or pin 8 of a DE-9) be connected to an appropriate signal. Others may require connections for Data Carrier Detect (DCD) (pin 8/pin 1) and Data Terminal Ready (DTR) (pin

COMPUTER INTERFACING

Nonstandard RS-232C Serial Ports

20/pin 4). You may jumper the computer's input and output signals on DB-25 pins 4/5 (DE-9 pins 7/8), and on DB-25 pins 6/8/20 (DE-9 pins 6/1/4) in addition to the connections shown in the "3 wire" RS-232 diagram above, or build an "8 wire" interface cable as shown below.

Jumpering RTS/CTS and DTR/DSR/DCD at the Computer Cable End

DB-25:

DE-9:

4 ----- 5	7 ----- 8
6 ----- 8 ----- 20	1 ----- 4 ----- 6

"Eight-wire" RS-232 Cable

Computer DB-25 or DE-9 to TINY-2/MICRO-2 DE-9P

1shell5
233
322
477
588
666
755
811 "connected" signal
204NC

Nonstandard RS-232C Serial Ports

NOTE: Some computers which require an external TTL to RS-232 adapter to support an RS-232 serial port may be interfaced to the TINY-2/MICRO-2 without the RS-232 adapter unit by using the TINY- 2/MICRO-2 TTL level signals.

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

- Voltage levels sent by the computer must be greater than +3 volts in one state and less than -3 volts in the other state.
- The polarity of the signals must conform to the RS-232C standard, i.e. the low voltage state is a logical "1" and the high voltage state is a logical "0".

Make or buy a cable that provides the following connections.

- The computer serial port common (ground) pin must be tied to the TINY-2/MICRO-2 serial port connector pin 5.
- The signal line that sends data from the computer must be tied to the TINY-2/MICRO-2 connector pin 3.
- The pin on which the computer receives data must be tied to the TINY-2/MICRO-2 connector pin 2.

The documentation provided with the computer or its accessory serial port should clarify any requirements for additional signals.

Serial Port Operation At TTL Levels

The TINY-2/MICRO-2 packet controller is designed to operate with computers which have a TTL level serial port without requiring the use of an RS-232C adapter. This includes such popular computers as the Commodore VIC-20, C-64, and C-128. TTL signals are routed through an 8 pin connector on the rear panel. TTL signal levels must be greater than +2.4 volts in one state and less than +.4 volt in the other state.

Use caution in connecting the TTL cable to insure that pin one of the cable is attached to pin one of the connector. Pin one is clearly marked on the rear panel of the controller. Pin one of the cable connector is marked with a red stripe. Be sure to observe this marking both in constructing and attaching the cable.

TTL serial port signals used by the TINY-2/MICRO-2 controller.

P3	Signal Name	Description
1	Receive Data	Serial data to computer from the TINY-2/MICRO-2.
2	Carrier Detect	Goes high when connection exists.
3	Frame Ground	Common ground for case and data lines.
4	Clear to send	Computer signal clearing controller to send data.
5	Transmit Data	Serial data to the TINY-2/MICRO-2 from computer.
6	Signal Ground	Common ground for both data lines and case.
7	Request to send	Controller signal to computer when data is ready.
8	No Connection	

Serial Data Signal Definitions

This section describes the pins used on the packet controller's serial port connectors. Since the pins are defined with respect to the Data Terminal Equipment (DTE) (computer) end of the circuit, and the packet controller is wired as Data Communications Equipment (DCE) (modem), some standard pin names appear to be the reverse of the packet controller function. A complete RS-232 signal table appears in the Appendices.

Serial Port Pin Functions

- **Frame Ground (FG)** is provided for attachment to the chassis of the packet controller and the chassis of the attached computer terminal. This is common to the signal ground on the TINY-2/MICRO-2.
- **Transmit Data (TXD)** is an input line to the packet controller on which the attached computer terminal device sends data.
- **Receive Data (RXD)** is an output line from the TINY-2/MICRO-2 on which it sends data to the attached computer terminal device.
- **Request to Send (RTS)** is an input to the packet controller signaling that the attached computer terminal device is ready to accept data from the TINY-2/MICRO-2. This line is used for hardware flow control.
- **Clear To Send (CTS)** is an output from the packet controller signaling the attached computer terminal device to send data to the TINY-2/MICRO-2. This line is used for hardware flow control.

- **Data Set Ready (DSR)** is an output from the packet controller telling the attached computer terminal device that the packet controller is operational. On the TINY-2/MICRO-2 this pin is tied high (ready).
- **Signal Ground (SG)** is the common, or return, path for all signals between the TINY-2/MICRO-2 and the attached computer terminal device. It is common with the frame ground.
- **Data Carrier Detect (DCD)** is an output from the packet controller. DCD reflects the status of the CON LED: It is high when a packet connection exists and it is low when no connection exists. This signal is useful when the packet controller is used with telephone style and BBS software, since the connect signal, like a modem carrier on the telephone, indicates the presence of a user.
- **Data Terminal Ready (DTR)** is an input to the TINY-2/MICRO-2 signaling that the attached computer terminal device is ready to accept data from the packet controller. Not used on the TINY-2/MICRO-2.

The table below provides a cross reference to the pinouts of the industry standard 25 pin connector as used on many computers and terminals, the TNC-2 and PacComm TNC-200, and relates them to the newer 9 pin connectors used on many modern computers and the TINY-2/MICRO-2. Note that connecting a DTE device to a DCE device using either 25 pin connectors or 9 pin connectors at both ends of the cable provides a 'straight through' wiring of pin numbers, i.e. connect column 1 to column 4 OR connect column 2 to column 3. However, when a 25 pin connector and 9 pin connector are used on the same cable, i.e. column 1 to column 3, several pairs of signal lines must be crossed.

TINY-2/MICROPOWER-2 RS-232 Serial Port Signal Cross Reference

	1	2	3	4
Conventional RS-232-	Conventional RS-232 DTE DB-25 Pin Nr.	IBM AT RS-232 DTE DE-9P Pin Nr.	TINY/MICRO RS-232 DCE DE-9P Pin Nr.	TNC-2(00) RS-232 DC DB-25S Pin Nr.
FG	1	shell	5	1
TXD	2	3	3	2
RXD	3	2	2	3
RTS	4	7	7	nc
CTS	5	8	8	5
DSR	6	6	6	6
SG	7	5	5	7
DCD	8	1	1 ^a	8 ^a
Test	9	V+	-	+12V
Test	10	V-	-	-12V
DTR	20	4	nc	20
RI	22	9	9 ^b	(23)

^a This line goes high when a connection exists

^b RFDCD (Also used for multi-node NET/ROM and ROSE)

RS-232C Handshaking Protocol

The CTS and RTS lines of the TINY-2/MICRO-2 serial port (labeled P4 on the circuit board and RS-232C on the rear panel) are used by the TINY-2/MICRO-2 firmware for hardware "handshaking" to control the flow of data between the computer and the packet controller.

The TINY-2/MICRO-2 always asserts (makes true) Data Set Ready (DSR) on P4 pin 6 via a pullup resistor. Thus, whenever the TINY-2/MICRO-2 is powered up, it signals to the computer connected to P4 that the packet controller is "on line."

The computer terminal indicates it is ready to receive data from the TINY-2/MICRO-2 by asserting its Request To Send (RTS) signal on P4 pin 4. The packet controller will send data when RTS is asserted (if there is data to be sent). If the computer is not ready to receive data, it negates (make false) RTS to the TINY-2/MICRO-2. Thus, data flow from the packet controller to the computer is controlled by the use of the RTS line.

The packet controller asserts its Clear To Send (CTS) output, P4 pin 5, whenever it is ready to receive data from the computer. If the TINY-2/MICRO-2's buffers fill, it will negate CTS, signaling the computer to stop sending data. The packet controller will assert CTS when it is again ready to receive data from the computer. Thus, data flow from the computer to the packet controller is regulated by the use of the CTS line. The CTS line is always toggled, even if "software flow control" is enabled in this direction.

If "software flow control" is preferred, or if the computer terminal serial I/O port does not implement CTS/RTS and DTR/DSR handshaking, then a three wire cable should be used. If these RS-232 control lines are not connected they will be pulled up (and thus asserted) by resistors at the TINY-2/MICRO-2 end and proper operation will result. However, a non-standard serial port may use some pins for other purposes, such as supplying power to a peripheral device, so *be sure that the system either implements the CTS, RTS and DSR handshake or has no connections to these pins of P4 whatsoever*. Note that reference to RS-232C "compatibility" or the presence of a DB-25 type connector does not guarantee that you have a standard RS-232C serial port!

The PacComm TINY-2/MICRO-2 packet controller supports asynchronous computer terminal baud rates of 300, 1200, 2400, 4800 and 9600 and 19,200. The port supports standard parity options as well as 7- or 8-bit character lengths. Factory settings are 1200 bauds, eight bit wordlength, no parity and one stop bit. (UK = 1200,7,E,1)

If you want to interface the TINY-2/MICRO-2 with a device configured as DCE, such as a telephone modem or another packet controller, a so-called "null modem" cable may be constructed to interchange the data and handshake signals.

RADIO INTERFACING

This section describes how to connect the signal cable between the TINY-2/MICRO-2 and the radio. The interconnection should be planned so as to minimize pickup of stray audio and RF noise by the lines. You should use shielded wire for all signal-carrying leads.

Read the remainder of this chapter carefully before starting to interface the packet controller to the radio.

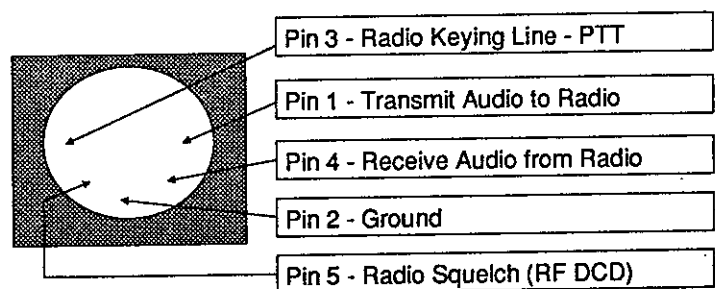
Connect the TINY-2/MICRO-2 and radio according to one of the following methods. Then turn on the packet controller and computer and start the computer communications program. Connect the radio to a dummy load and listen to the transmission with another nearby radio. Set the transmit level by the procedure in the Adjustments Chapter.

You should use shielded wire for all signal-carrying leads. If you notice a significant hum level in the monitored audio, take measures to remove it. If the transmitter has an adjustable microphone gain control, try reducing the sensitivity of the transmitter microphone circuit and increasing the signal level from the TINY-2/MICRO-2 to minimize hum or other noise problems.

Radio Connector (J2) Pin Definitions

Interfacing the TINY-2/MICRO-2 to a radio involves connecting the following signals at J2, the five pin DIN connector.

- Pin 1 - Audio output from the TINY-2/MICRO-2 to the transmitter.
- Pin 2 - Ground for both audio and PTT.
- Pin 3 - Push-to-talk to allow keying the transmitter.
- Pin 4 - Receive audio from the receiver speaker or auxiliary jack to the TINY-2/MICRO-2.
- Pin 5 - Optional Radio Squelch input for RF Carrier Detection.



Radio Connector Viewed from Rear of Packet Controller

Direct Connection to Microphone and Speaker

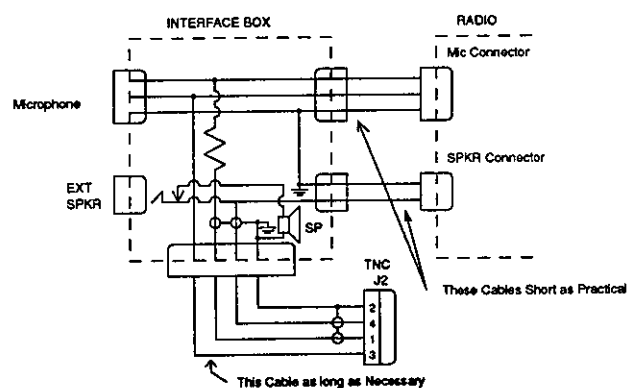
The packet controller was designed to allow hookup and testing to be done without any modifications to the radio or any signal level balancing devices in the cables. The direct interconnection method is useful if you devote the radio exclusively to packet. If you share the radio between packet and voice, you may want to use the external interface method below. The packet controller's audio signals are fed directly into the microphone connector or similar auxiliary input of the radio, and the TINY-2/MICRO-2 output is adjusted to give a proper modulation level. The receiver audio is taken from an auxiliary audio output or speaker jack and fed directly to the packet controller.

Accessory Jack Connection

If the radio has an accessory jack with PTT, transmit audio, and receive audio signals, all connections between the radio and TINY-2/MICRO-2 may be done through this jack. The TINY-2/MICRO-2 is designed to operate with from 10 to 700 millivolt audio input. This is compatible with the signal level provided by most modern transceivers at the "Fixed Audio Output" connector on the rear panel. It is recommended that a squelched fixed audio output be used if available. If the audio output is squelched or can be adjusted to a low enough level to prevent false carrier detection, a direct connection to the TINY-2/MICRO-2 audio input may be made.

External Interface Box.

If you desire to use the radio for both packet and voice, and the radio does not have an accessory jack and you don't wish to add a connector to the radio, you should construct a separate interface box to permit simultaneous connection of the TINY-2/MICRO-2 and a microphone. It may either have its own speaker or pass the signal through to a speaker so that you can monitor the receive audio while the packet controller is hooked up. A schematic of an external interface box is shown below.



External Interface Circuit Schematic

HARDWARE DESCRIPTION

This section includes detailed hardware specifications and a functional description of the hardware design of the TINY-2 and MICROPOWER-2 packet controllers. Specifications apply to both packet controller models unless specified otherwise. MICROPOWER-2 specifications (when different) will be placed in square brackets []. Firmware specifications and operating instructions are contained in the Operating Manual.

LEDs

PWR

The POWER LED is illuminated whenever the TINY-2/MICRO-2 is connected to a power source and the power switch is on.

CON

The CONNECT LED illuminates when an AX.25 connection exists on the selected stream for that port. (See multiconnection explanation in Operating Manual).

STA

The STATUS LED illuminates whenever an AX.25 frame has been sent but not yet acknowledged. If a connection is to be terminated (DISCONNECTED) and the STA LED is lit, some frames which were sent but not acknowledged may be lost.

PTT

The PTT LED illuminates whenever the TINY-2/MICRO-2 activates the radio keying line.

DCD

The DCD LED illuminates when the modem senses a signal.

Detailed Circuit Description

- **Oscillator:** The crystal, capacitors and resistors in the 74HC4060 (U7) circuit provide for a stable clock source for the 4.9152 MHz master clock. The 74HC4060 also provides all baud clocks (X16) and a 600Hz signal for the real time clock interrupt applied to the SIO (U14) SYNC-B input.
- **CPU Complex:** The microprocessor (U1) is an NMOS Z-80 (8400-06) [CMOS, 84C00-06]. Both asynchronous serial and HDLC functions are performed by a Z80 SIO/0 (8440-06) [84C40-06] (U14). The 27256 [27C256] EPROM (32k bytes) (U2) contains all the firmware, addressed from 0 to 7FFF. The 32k x 8 RAM (U8) holds all buffered data and battery backed parameters for the 8400-06 [84C00-06] (Z-80) CPU (U1). RAM is addressed from 8000 to FFFF. Each memory device's chip select is provided by the 74HC139 decoder (U3), and the RAM chip select is also buffered by a power failure detection circuit. When voltage is low or removed, the RAM enters low power mode, drawing approximately one microamp while retaining memory contents.

HARDWARE DESCRIPTION

Detailed Circuit Description

- **Serial Port:** SIO (U14) channel B performs asynchronous communications through either a MAX 231 single chip CMOS RS232 transceiver (U15), or at TTL levels, via an 8-pin connector (P3). RS232 signals are interfaced via an "IBM AT" style DE-9P connector with a DCE pinout. Baud rates of 300, 1200, 2400, 4800, 9600, and 19.2k are supported.
- **HDLC:** The HDLC function is performed by SIO (U14) channel A connected to the modem disconnect header (J5), and to the onboard integrated circuit modem. Data rates of 300 bauds to 19.2 kilobauds are supported at the header. The SIO HDLC data signal is converted from NRZ to NRZI in two logic stages enroute to the modem.
- **Watchdog Timer:** The SIO (U14) RTSA (pin 17) provides the signal that ultimately keys the radio transmitter. When the TINY-2/MICRO-2 is transmitting, RTSA goes low, which is fed to a Schmidt trigger gate (U6 pin 9) which turns on the PTT LED and returns a high signal to the gate of a VN10 (FET) transistor (Q2). This high state on the VN10 gate causes it to conduct to ground, keying the transmitter, and starts C26 charging through R17. If the RTSA signal remains low more than 60 seconds, C26 reaches the Schmidt trigger voltage, causing the gate (U6) to turn off the PTT LED and the VN10, thus unkeying the transmitter.
- **Modem:** The TCM3105 modem (U16) employs a Bell 202 modulator and demodulator in a single CMOS package. The modulator transmits either 1200 or 2200 Hz depending on the state of the SIO TXD line (U14 pin 14). The modem tone output (U 16 pin 11) goes through a 10K trimpot to allow amplitude adjustment to achieve proper transmitter deviation.

Since the TCM 3105 always produces tones when powered, there is an audio cutoff circuit employed using a VN10 FET (Q3) and a 2.2k resistor that swamps most transmit audio to ground when the RTSA line is high (receive). This allows normal voice use of the radio while the TINY-2/MICRO-2 is connected.

The demodulator contains a switched capacitive filtering network, and is sensitive to audio levels down to 10 mv. The receive bias adjustment (R9) allows optimizing the bias level for a particular TCM3105 modem IC. Received data is output on U16 pin 8 and is routed through the clock recovery circuit before the SIO (U14).

NOTE: The on-board radio modem is only capable of 1200 baud operation as installed. For higher (or lower) radio data rates, attach an external modem to the J5 modem disconnect header.

- **Battery backed RAM (bbRAM):** The entire 32K of RAM (U8) is backed up with a 3 volt lithium battery. The 74HC132 IC (U3) is also battery powered keeping pin 20 (CE) of the RAM disabled (high) during periods of low power to the TINY-2/MICRO-2 (typically 7.5 volts and below).

- **Power Supply:** The 7805 regulator (U5) supplies the main portion of the board with clean, stable 5 volt power. The incoming positive voltage is routed through a choke (L1) and diode (D6) for reverse polarity protection and to the power switch. It feeds the 7805 regulator and pin 14 of the MAX 231 IC (U15). The MAX 231 has internal oscillators which allow it to generate a negative voltage to support the RS232 levels. A voltage approximately one volt below the incoming supply voltage can normally be measured. The battery circuit is isolated by diodes to keep the RAM and the 74HC132 powered whenever JP-B is installed. The TINY-2 requires approximately 220 ma at 9-14 vdc [MICROPOWER-2 = 40 ma.]

Modem Disconnect - J5

The modem disconnect on the TINY-2/MICRO-2 PC board (J5) is provided for using higher-speed modems, or more sophisticated, higher-performance modems for satellite or other uses. This connector is the same as the TAPR TNC-2 header.

A physical connector for J5 is installed on the MICROPOWER-2, but not on all versions of the TINY-2. Any standard 20-pin header for use with IDC cable connectors should be suitable for installation on the TINY-2. Parts are available from PacComm. 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.

To use an external modem, it is necessary to cut the traces between pins 1-2 and 17-18 on the bottom of the TINY-2/MICRO-2 PC board with a sharp knife or MotoTool. If RX Clock is provided by the external modem, then cut 13-14. If CTS is provided, cut 9-10.

CAUTION: DO NOT CUT ANY OTHER TRACES WHICH MAY PASS THROUGH THE J5 AREA AND WHICH ARE NOT CONNECTED TO J5 PINS!

Once the traces are cut at J5, you must install push-on jumpers or a shorting connector on the above pin pairs to use the on-board modem. This also applies to the MICROPOWER-2.

The signals used at connector J5 are at standard TTL interface levels. A TTL high is greater than +2.4 volts but less than +5.25 volts. A TTL low is less than 0.4 volts but greater than 0 volts.

DO NOT connect an RS-232C level modem directly to J5!

If you elect to use an off-board modem, be sure to properly shield the interconnecting cables for RFI protection.

J5 Pin Definitions

The following information is primarily for those who wish to interface external modems to the packet controller. Familiarity with modem and serial data channel terms is assumed.

Pin 1 Carrier Detect Input

This pin tells the SIO 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 normally jumpered to pin 2 when the on-board modem is used.

Pin 2 Carrier Detect Output

This pin is an output from the on-board modem (via an inverter gate) and provides the necessary signals required by pin 1 above. It is normally jumpered to pin 1 when the on-board modem is used.

Pin 3 SIO Special Interrupt Input

This pin is routed to the radio port DCD input pin on the SIO. This pin is normally jumpered to pin 4 when the on-board modem is used.

Pin 4 SIO Special Interrupt Generator Output

This signal is an output from the on-board modem. It is normally jumpered to pin 3 when the on-board modem is used.

Pin 5 SIO RTS Output

This signal is used for transmitter activation. It activates the PTT pin of the radio connector via the watch-dog timer. The SIO will pull this output low when the TINY-2/MICRO-2 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 CONNECT Status Output

This pin is an output from the SIO. It is normally low and goes high only when the TINY-2/MICRO-2 is in a connected state. Its status is reflected by the CON LED.

Pin 8 Unacknowledged Packets Pending Status Output

This pin is an output from the SIO. It is normally low and goes high only when the TINY-2/MICRO-2 has unacknowledged packets in its transmit buffer. Its status is reflected by the STA LED.

Pin 9 SIO CTS Input

This pin is an input to the SIO. 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 packet controller 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 SIO when it is ready to accept data. This pin is normally jumpered to pin 9.

Pin 11 Transmitter Clock (16x) Input

This pin is tied to the NRZ-to-NRZI converter, which expects a clock signal at 16 times the desired radio port data rate, e.g. 4800 Hz for 300 bauds. This pin is normally jumpered to pin 12 when the on-board modem is used.

Pin 12 Transmitter Clock (16x) Output

This pin is tied to the radio baud rate switch network. It provides a clock at 16 times the desired radio port data rate. This pin is normally jumpered to pin 11 when the on-board modem is used.

Pin 13 Receive Clock Input

This pin is tied to the SIO receive clock input pin. It expects a clock at the desired data rate (1200 Hz for 1200 bauds), of the proper phase relationship to the received data. This pin is normally jumpered to pin 14 when the on-board modem is used.

Pin 14 Receive Clock Output

This pin is the received data clock signal derived from the NRZI-to-NRZ converter. This pin is normally jumpered to pin 13 when the on-board modem is used.

Pin 15 TINY-2/MICRO-2 Ground Reference

This pin ties to the TINY-2/MICRO-2 digital ground at the SIO.

Pin 16 Additional Ground Reference**Pin 17 Receive Data Input**

This pin is the received data input to the NRZI-to-NRZ converter. This pin is normally jumpered to pin 18 when the on-board modem is used.

Pin 18 Receive Data Output

This pin provides receive data from the on-board modem. This pin is normally jumpered to pin 17 when the on-board modem is used.

Pin 19 Transmit Data Output

This line is the SIO 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 transmitted by the modem. This pin is normally jumpered to pin 19 when the on-board modem is used.

ADJUSTMENTS

Jumper Functions

The following table lists the function of each jumper on the TINY-2/MICRO-2 PC board. The normal (standard jumper setting) positions for each are shown.

JP B : Lithium Battery

- ON (normal) Lithium battery connected
- OFF Lithium battery disconnected

JP D: NET/ROM + RFDCD

- N (normal) NET/ROM input to DCDB
- RF RF Squelch to DCDB

JP L: Digital Loopback

- N (normal) Normal Modem Operation
- L Digital loopback mode

JP RA: Audio Input Impedance

- ON (normal) Speaker audio input
- OFF High impedance audio input

JP Aux: Auxiliary I/O

- Auxiliary I/O lines for MICROPOWER-2 with printer option

JP 8: Power Connection

- Auxiliary connection point for 12v and gnd for accessories

Trimpot/Varicap Adjustments

Most of the following adjustments must be performed with the TINY-2/MICRO-2 removed from its cabinet. Remove the packet controller from its cabinet by removing both front and rear end plates and plastic bezels. Use caution not to damage the grounding straps attached to some of the end plate screws.

- C17 -- CPU oscillator frequency
- R9 -- Modem bias level
- R12 -- Transmit output level
- R26 -- Modem carrier detect level

Baud Rate Setting

To set the baud rate on the packet controller, remove the TINY-2/MICRO-2 from its cabinet. There are two sets of jumper headers [DIP switches] along the front edge of the circuit board with labels of 3, 12, 24, 48, 96, and 19.2 for both TERMINAL and RADIO. These numbers correspond to 300, 1200, 2400, 4800, 9600, or 19.200 baud rates. Place the jumper on the two pins [turn the section of the DIP switch ON] for the desired rate for both the computer and radio.

WARNING: Place one and only one jumper for the computer side, and only one on the radio side. [Only one switch on the MICROPOWER-2 should be ON per section.]

Audio Input Level

The TINY-2/MICRO-2 is designed to operate with from 10 to 700 millivolt audio input. If you obtain audio from the headphone or external speaker jack on the radio, place a shorting block on the pins of JPR (factory default) and adjust the audio level to approximately 20 mv with the aid of a high impedance voltmeter or oscilloscope. If no such measuring equipment is available, set the volume control at a position below that normally used.

If direct connection is made to the radio's discriminator (demodulator) or any other high impedance audio source, remove the shorting block from JP R.

NOTE: The audio level should be set as to as low a value as possible if an unscquelched radio is used. The TINY-2/MICRO-2 modem will show false DCD indications on noise if the audio level is set too high.

Transmit Level Setting

Connect the TINY-2/MICRO-2 and radio as shown in the radio interfacing chapter. Turn on the packet controller and computer and start the computer communications program. Connect the radio to a dummy load and listen to the transmission with another nearby radio. The output level is set by adjusting trimpot R12 which is accessible through the rear panel opening marked "TTL Computer."

1. Enter the command **CAL** <CR> and the letter "K" to cause the TINY-2/MICRO-2 to key the radio with a steady tone. The high or low tone is selected by pressing the space bar. **NOTE:** The watchdog timer will cause the PTT circuit to reset approximately one minute after the 'K' key is depressed. If you need a longer adjustment time, type 'Q' <CR> to return to the command prompt and restart the procedure.
2. With the TINY-2/MICRO-2 keying the transmitter and the higher frequency tone selected, adjust the transmit audio level trimpot (R12) while observing a service monitor or deviation meter tuned to the transmitting frequency. Set the deviation for 3 to 3.5 KHz on the higher pitched tone and check that the lower tone is at least 2.5 KHz deviation.

3. If the proper equipment is not available as described above, make the deviation adjustment by turning the adjustment screw on the trimpot until no increase in output level is heard at the monitoring receiver. 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.

NOTE: It is better to have too low than too high a drive level. A high drive level causes over-deviation, unnecessary interference, and makes your signal less copyable. Listeners may also notice some low-level packet tones when using the radio for voice work.

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

Modem Receive Bias Adjustment

Trimpot (R9 100K) is the modem bias adjustment. The factory procedure below is used to adjust the modem bias. It normally results in a value of 2.65 to 2.70 vdc on pin 7 of the TCM3105 modem chip (U16). The setting does not seem to drift over time but may need to be changed if a different modem chip is substituted. If it is necessary to adjust the modem bias, the following procedure should be used.

1. Prepare to perform an audio loopback by jumpering pins 1 and 4 of J2, the radio connector.

2. Give the commands: **CAL** <CR>, **KD** <CR> to provide a 'dithered' transmit signal for loopback demodulation. When the adjustment is completed, type 'Q' to discontinue the tone. NOTE: The watchdog timer will cause the PTT circuit to reset approximately one minute after the 'KD' keys are depressed. If you need a longer adjustment time, type 'Q' <CR> to return to the cmd: prompt and restart the procedure.

3. Observe the signal output on pin 8 of the TCM3105 (U16) with an oscilloscope. It may be necessary to increase the modem output level by adjusting trimpot R12 to get a strong enough signal for decoding. The signal should be a square wave. Adjust R26 to cause the positive and negative portions of the square wave to be of equal duration.

4. Enter "Q", remove the jumper from J2 and readjust R12 if needed.

Carrier Detect Level

Trimpot R26 should be adjusted to cause the DCD LED to illuminate when a weak input signal to the modem is present and to go out when no signal is present. See the section on audio input levels.

Crystal Oscillator Adjustment

You may adjust the oscillator with the variable capacitor C17, to within a 5 KHz range (typical) from its 4.9152Mhz nominal frequency to move a harmonic a few kilohertz off of a favorite operating frequency. The setting of this frequency is not critical. A 30khz error at the oscillator frequency will result in only a .6% error at 1200 baud.

TROUBLESHOOTING

WARNING: Never remove or insert an IC with power on!

The PacComm packet controller 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 above. 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.

Press all socketed ICs into their socket along each edge. If there is any sign of corrosion on any IC pins, remove each IC, one at a time, and burnish the pins. Often simply removing and reinstalling the ICs will cure intermittent problems caused by vibration or humidity affecting the IC sockets. Inspect all solder joints for a smooth shiny appearance. A dull gray appearance may be an indication of a crystalized solder joint. Use a small soldering iron designed for printed circuit work and resolder the connection.

Remember that all the logic circuits operate at standard TTL levels (a "low" is less than plus 0.4 V and a "high" is greater than plus 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 (U2 and U8 pins 20), the IOREQ line on CPU (U1) pin 20 and the INT line on CPU (U1) 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 defective solder joint. Follow the instructions above for inspecting solder joints. 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.

If an oscilloscope is not available, then a voltmeter can help you find serious difficulty on digital signal and address lines. Set the meter for the scale where a full 5 volts can be displayed. A reading of 1.5 to 3.5

volts indicates some varying digital signal. A reading of over 3.5 volts or under 1.5 volts is most likely an indication of trouble. A very low, or very high voltage reading is a sign of no activity, or open and shorted circuits. If you cannot pinpoint the trouble using this set of guidelines, then you must either obtain an oscilloscope, or send the board back to PacComm 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.

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.

Loopback testing

Loopback testing consists of having the transmit section of the packet controller communicate with the receive section. If this procedure is successful, the correct operation of both transmit and receive sections can be verified. If data is not successfully passed, clues to the malfunction may be obtained.

There are two types of loopback testing, analog and digital. For both types of test, the command FULLDUP must be ON and some type of connection must be made between the transmit output and the receive input. Once the proper arrangements are made, testing consists of attempting to pass information from the transmit section to the receive section of the packet controller. One method is to send data to and from the PMS (if the PMS option is installed) by connecting to MYP-CALL. You may also connect to MYCALL and send and receive from the terminal, or send UI frames and monitor them. Be sure to undo all the loopback steps once the testing is completed.

Analog loopback

Remove JPR to place the input circuit in a high impedance state. Install a homemade jumper between pins 1 and 4 of the radio connector J2. Send UI frames and monitor for their reception or try a connection as described above. The output level trimpot R12 may have to be turned to near maximum to allow the modem to hear itself. Failure of the analog loopback test does not necessarily mean there is a modem problem since the transmit circuits and receive circuits are not optimized for direct connection.

Digital loopback

Move the JP L jumper to 'LOOP' and attempt to pass UI frames or connect. Digital loopback is very reliable and should work every time if there is no problem in the digital portion of the board.

General Tests

The TINY-2/MICRO-2 has status LEDs which are powerful tools in troubleshooting. Many suspected problems may be disproved by the following simple tests. Connect a power source to the TINY-2/MICRO-2 and a voltmeter (15 volt scale) to pin 2 of the RS-232 connector. Do not connect any computer or radio cables. Turn on the TINY-2/MICRO-2. Observe the LEDs. If the TINY-2/MICRO-2 powers up with the PWR, CON and STA LEDs lit, followed by the CON and STA LEDs extinguishing a second or so later, and the voltmeter shows a lot of activity for about 1/2 second, the TINY-2/MICRO-2 digital section is operating and attempting to send a sign on message out the serial port.

Step 1: Obvious Problems

Look for any unusual physical symptoms. Is the unit drawing excessive current? Are any components discolored? Does something smell burnt? Do any of the parts seem excessively warm? In general, their normal temperature (NMOS) will be quite warm to the touch. CMOS parts should not show any heating.

Step 2: Power Supply

The first thing to check in any malfunction is the power supply.

NOTE: A frequent problem is the use of a poorly filtered or undervoltage DC power supply.

Check that the voltage under load is not low and that excessive ripple is not present. Check the power supply levels at the output of the voltage regulator. 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.

Computer Terminal Interface

If you can't get the packet controller to sign on and accept data from the terminal or computer, the problem may be in the serial interface.

NOTE: The primary cause of non-communication between the packet controller and the computer is improper cabling setup.

Inspect the interconnection cabling. Does it work on another controller or RS-232 device? Has the radio and/or computer been successfully used on packet with this or another packet controller? Are all the connections tight? Has the cable frayed or broken?

Packet controller won't sign on to the computer.

If you find the TINY-2/MICRO-2 won't send data to the computer, but appears to operate properly as described in General Tests above, verify that the CTS line at RS-232 connector P4 pin 8 or TTL connector P3 pin 4 is not being held low. The packet controller will not send data to the computer unless its CTS line is asserted. If the computer does not implement the RTS/CTS protocol, the RTS/CTS lines (pins 7 and 8 on P4) should remain unconnected to the computer or else tied together.

Check the baud rates set on the packet controller and computer terminal. Some mismatched combinations of baud rate on controller and terminal will cause nothing to be displayed.

If the above checks are ok, observe pin 26 of the SIO (U14) with an oscilloscope and cycle the packet controller power switch. Transitions on this pin shortly after reset indicate that the packet controller is sending data. If no signal transitions are seen, recheck that the CPU appears to be operating (see below.) If data appears to be present, verify that TTL transitions are also present on the Max 231 (U15) pin 8 and RS-232 signal transitions are present on Max 231 (U15) pin 11 and P2 pin 2.

The controller prints only gibberish.

This indicates that some combination of the data rate (terminal baud rate), parity option, or number of start and stop bits are not set the same at the packet controller and at the computer terminal. Verify that the computer terminal is set for eight data bits, no parity, and 1 stop bit. These settings may be made with DIP switches, jumper pins, or software commands depending on the computer terminal in use. The above settings are the default sign-on settings stored in the TINY-2/MICRO-2 EPROM. Perform a hard reset by pressing the power switch OFF then ON (out then in). The sign on message should appear.

If the packet controller's sign-on message lines appear to be typed over each other, you should set the packet controller command **AUTOLF ON**. If the sign-on message appears to be double-spaced, you should set **AUTOLF OFF**. If the sign-on message appears to be single-spaced, **AUTOLF** is correct.

The controller won't accept commands.

After the packet controller 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 SIO (U14) pin 23 has a voltage level between 0 and +0.4 volts on it. The voltage on Max 231 pin 5 should be greater than +3 volts. If these conditions are not met, check the cable wiring and refer to the section about terminal interface troubleshooting.

If the above checks are OK, use an oscilloscope to verify that data is present on SIO (U14) pin 28 and Max 231 (U15) pin 10 when you strike a key on the computer. If not, the data isn't getting from the computer to the packet controller. Check the serial connector in use (P4-RS-232,

TROUBLESHOOTING

Other symptoms

or P3-TTL), the cable and Max 231 (U15) again. Finally, be sure that the computer actually uses levels less than -3 volts and greater than +3 volts for signal levels. 0 and +5 volts will not work unless the TTL port is used.

Other symptoms

TINY-2/MICRO-2 appears dead

If no LEDs wink during the startup 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 output of U7 (74HC4060) pin 9. The clock signal is a (possibly distorted) square wave signal. Also check for clock signals Z80 CPU (U1) pin 6 and SIO (U14) pin 20.

Verify that the battery backed-up RAM protection circuit is working by measuring that pin 28 of the RAM (U8) at least +4.7 volts after input power is applied.

If the unit is operating and accepting commands from the terminal, an overall check of the CPU complex may be conducted using the HEALTH command (see Operating Manual.)

Modem won't key transmitter.

Check to see the commands XMITOK and TXUIFRAM are ON. If the transmitter doesn't key, the problem is most likely in the connecting cable, or a carrier present (DCD) indication. If DCD is active and the command FULLdup is off, the controller will not activate the keying line. If this is the problem, see the DCD section below. If the connections appear to be in order, check 8440 (U14) 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 74HC14 (U17) pin 9 to U17 pin 8 (now inverted) to the gate of the keying FET(Q2). The FET output line changes from high impedance to low when a keying signal is present. This can be checked with an ohmmeter.

Uncopyable transmitted packets

If other stations seem unable to decode your packet transmissions, it is possible that your transmitter is being overdriven. Listen to the sound of your TINY-2/MICRO-2 transmissions on another receiver. Extremely weak signals indicate inadequate modem output. Refer to the Transmit Level Adjustment Procedure. You may also be able to hear the beginning of transmissions clipped or weak if inadequate transmit delay (RXD command) is being used for your transmitter. You can confirm that the digital logic and modem on the packet controller are operating properly by performing an Analog Loopback. If the loopback works properly, then re-set the output level taking care to set the drive level lower than previously. Note that direct connection to typical microphone inputs requires the drive trimpot to be turned to near the minimum signal position for proper signal levels. If the Analog Loopback fails to operate properly, perform a digital loopback. If this test operates correctly, the CPU, software, and SIO are operating correctly and the modem circuit is suspect.

Uncopyable received packets

If some but not all packets are copied, set the command MCOM ON. Many packet transmissions are control frames and contain no printable information. MCOM will cause these frames to be displayed. Confirm proper wiring of the radio cable by opening the radio squelch and raising the volume level with some type of signal tuned in. If the DCD LED flickers or solidly illuminates then receiver audio is reaching the packet controller modem. Test the modem by performing an audio loopback. If the test fails, perform a digital loopback to be sure the problem is really in the modem section. Listen to the audio signal from the radio applied to the packet controller or examine the signal with a scope for hum or other noise which might make the packet signals uncopyable. Confirm the proper level of the audio signal is applied to the packet controller with a digital meter or scope.

DCD LED flickers excessively or stays on

If the DCD LED flickers on and off excessively or remains constantly on only 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 audio is being obtained from a speaker output or other low impedance output, place a jumper on JPR located near R12. If high impedance audio is being used, JPR should be off. If these areas all check out, consider changing the value of resistor R10 to a lower value between 10 and 100 ohms.

Transmitter locks in transmit condition

The TINY-2/MICRO-2 is protected against PTT malfunction by a watchdog timer on the keying line. However, if unshielded, improperly grounded cabling is used to feed the transmitter key line, and other signal cables, RF energy can get back into the packet controller and lock up the keying transistor. Precautions are especially necessary if a high power amplifier is being used and there is a strong RF field in the area of the packet controller.

APPENDICES

Node Interconnection Cables

These diagrams are intended for use with NET/ROM, ROSE and other network EPROMS. If these diagrams do not agree with those suggested by the network software manual, use those provided by the software manual, modified as necessary for the TINY-2/MICRO-2 9 pin connector.

Dual TINY-2/MICRO-2 Node

Pin # Unit 1	Pin # Unit 2
2 -----	3
3 -----	2
7 -----	8
8 -----	7
5 & 9 -----	5 & 9

Note: Lines 7 and 8 are optional.

Dual Node, TINY-2/MICRO-2 and TNC-2(00)

Pin # TINY/MICRO	Pin # TNC-2(00)
2 -----	2
3 -----	3
7 -----	5
8 -----	20
5 & 9 -----	10 & 23

Note: Lines 7 & 8 to 5 & 20 are optional.

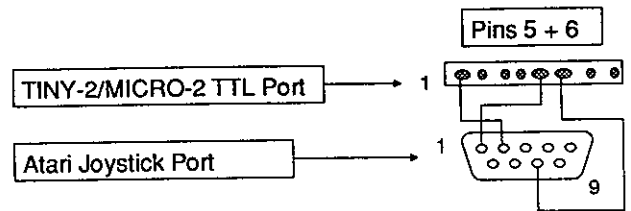
Triple Node

Follow the cable assembly instructions in the NET/ROM manual substituting pin numbers according to this table.

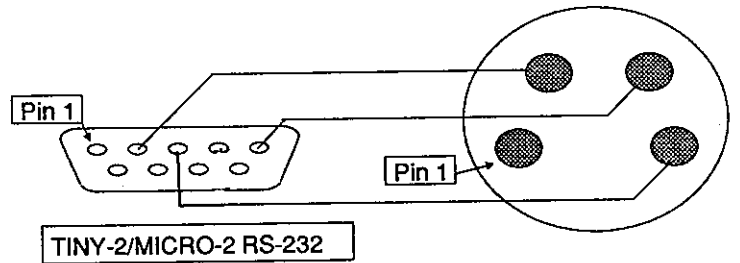
TINY-2/MICRO-2	TNC-2(00)
2-----	3
3-----	2
5-----	1 & 7
7-----	20
8-----	5
9-----	23

Triple (and larger) node cable diode matrices may be easily constructed using the "TNC Octopus" diode matrix board by John Painter. The Octopus is available from PacComm.

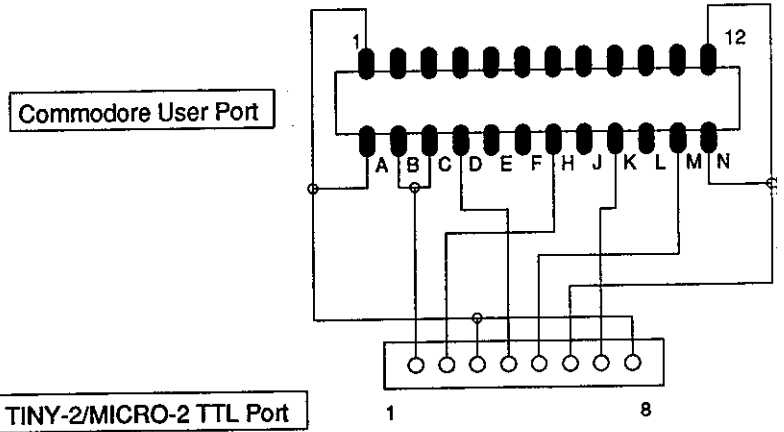
Computer Interfacing Diagrams



Atari Joystick Port to TINY-2/MICROPOWER-2 TTL Port

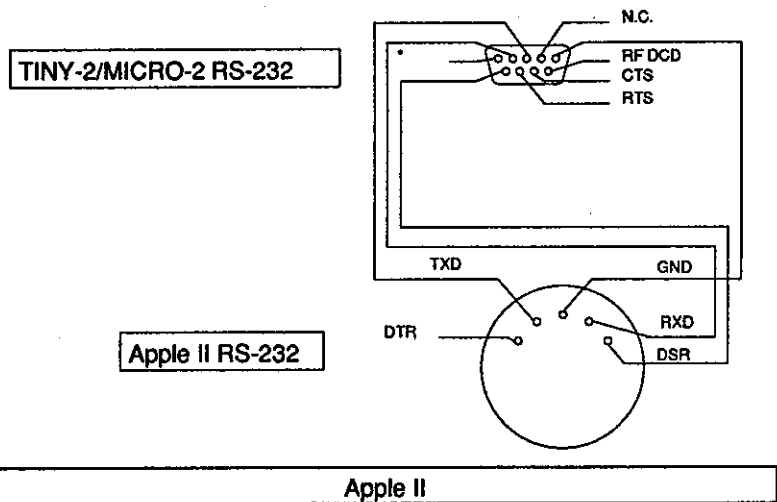
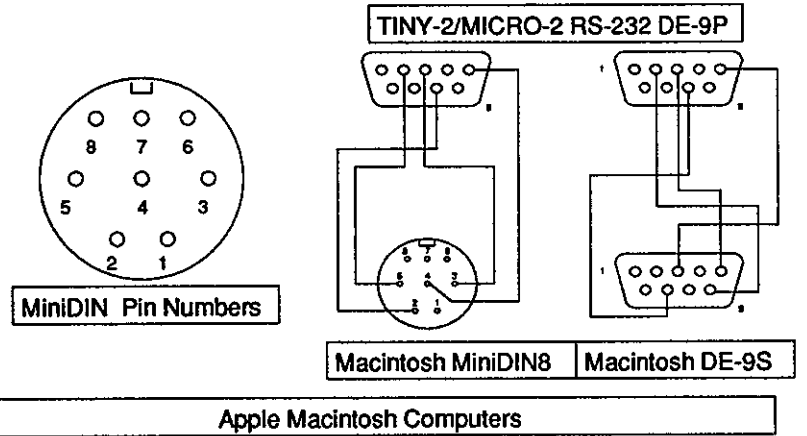


Tandy/Radio Shack Color Computer to TINY-2/MICRO-2 RS-232 Port



Commodore VIC-20/C-64/C-128 User Port to TINY-2/MICRO-2 TTL Port

Computer Interfacing Diagrams

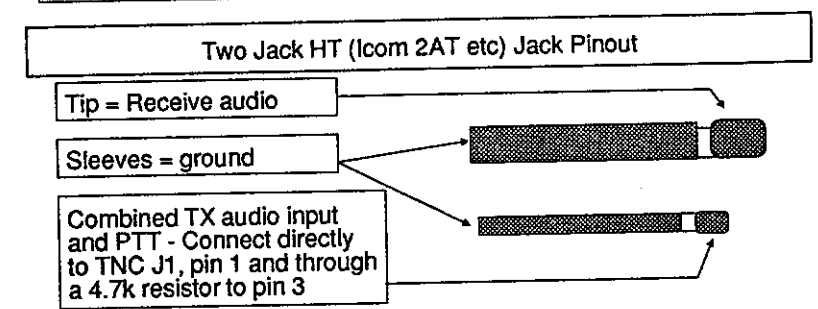
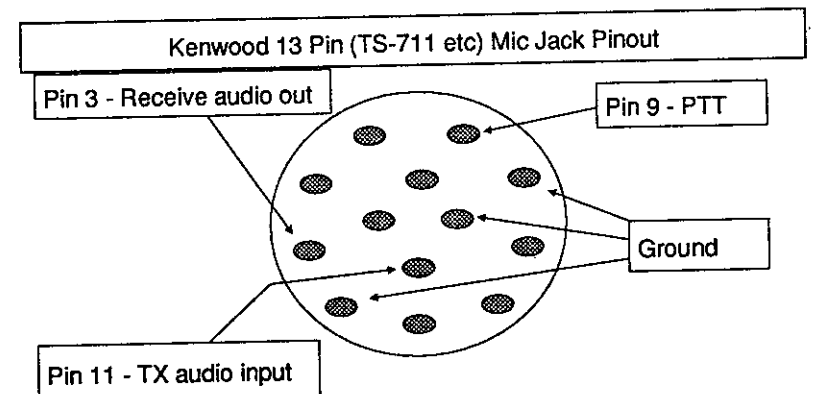
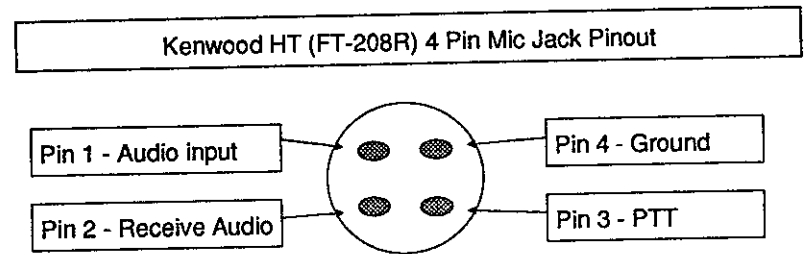
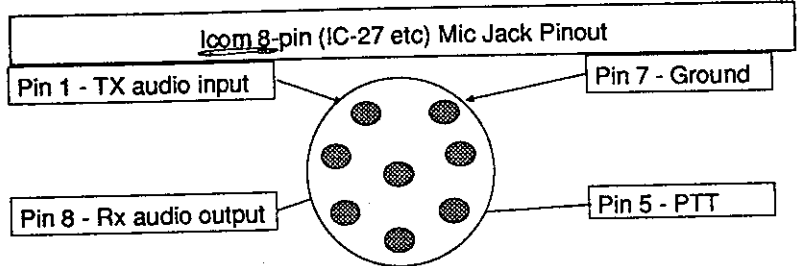


Color Computer TINY-2/MICRO-2 RS-232

- 2 ----- 2
- 3 ----- 5
- 4 ----- 3

Tandy/Radio Shack Color Computer

Radio Interfacing Diagrams



RS-232 Signal Table

The EIA RS-232C Standard

The Electronic Industries Association (EIA) RS-232C standard defines the interfacing between data terminal equipment and data communication equipment employing serial binary data interchange. Electrical signal and mechanical aspects of the interface are well specified. The complete RS-232C interface consists of 25 data lines. This would seem to be enough signals for a complex parallel communication line, but many of the 25 lines are very specialized and few are undefined. Most computer terminals only require from 3 to 5 of these lines to be operational.

Because RS-232C was originally intended for data communications equipment, a few lines must be swapped between a computer and terminal if no modem or other data communications equipment is used. These wires are swapped within the cable connecting the two devices. This line is called a null modem cable.

Pin Description

- 1 Protective Ground
- 2 Transmitted Data
- 3 Received Data
- 4 Request to Send
- 5 Clear to Send
- 6 Data Set Ready
- 7 Signal Ground (Common Return)
- 8 Received Line Signal Detector
- 9 (Reserved for Data Set Testing)
- 10 (Reserved for Data Set Testing)
- 11 Unassigned (See section 3.2.3)
- 12 Sec. Rec'd. Line Sig. Detector
- 13 Sec. Clear to Send
- 14 Secondary Transmitted Data
- 15 Transmission Signal Element Timing (DCE Source)
- 16 Secondary Received Data
- 17 Receiver Signal Element Timing (DCE Source)
- 18 Unassigned
- 19 Secondary Request to Send
- 20 Data Terminal Ready
- 21 Signal Quality Detector
- 22 Ring Indicator
- 23 Data Signal Rate Selector (DTE/DCE Source)
- 24 Transmit Signal Element Timing (DTE Source)
- 25 Unassigned

Bit-Shifted ASCII Character Table

The characters of AX.25 callsigns are standard 7 bit ASCII characters shifted to occupy the upper 7 bits of the byte. The following table contains the hexadecimal values for the characters of the alphabet and digits 1 - 0 bit shifted one place for use in callsign definitions in packet controller EPROMs.

Format:

Character
ASCII Hex Bitshifted Hex

A	B	C	D	E	F	G
41 82	42 84	43 86	44 88	45 8A	46 8C	47 8E
H	I	J	K	L	M	N
48 90	49 92	4A 94	4B 96	4C 98	4D 9A	4E 9C
O	P	Q	R	S	T	U
4F 9E	50 A0	51 A2	52 A4	53 A6	54 A8	55 AA
V	W	X	Y	Z	space	
56 AC	57 AE	58 B0	59 B2	5A B4	20 40	
1	2	3	4	5	6	7
31 62	32 64	33 66	34 68	35 6A	36 6C	37 6E
8	9	0				
38 70	39 72	30 60				

Example: Normal hex values for W1AW = 57 31 41 57

Bit-shifted hex values for W1AW = AE 62 82 AE

PacComm Manual Errata and Software Release Notes, March 5, 1992

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.

ALL PRODUCTS:

Power connectors are center pin positive.

Current operating instructions for all PacComm amateur products except the PC-100 series is Operating Manual for PacComm Packet Controllers, 4th Ed., 3rd Printing, October, 1991.

The KISS mode ignores DCD and acts as if FULLDUP is ON regardless of FULLDUP setting.

TINY-2 and MICROPOWER-2:

Current Firmware: 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 Ref. Manual for the TINY-2 and MICROPOWER-2, 2nd Ed., 2nd Prtg., Feb. 1991. 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.

PC-320:

Current Firmware: PC-320	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)

TNC-320:

Current Firmware: TNC-320	P1.1.6D4 PMS 3.0 \$65
European TNC-320	E1.1.6D4 PMS 3.0 \$6A

Technical Reference Manual for the TNC-320, First Ed., Second Printing, June 1991.

HandiPacket:

Current Firmware: HandiPacket	P1.1.6D5 PMS 3.0 \$38
European HandiPacket	E1.1.6D4 PMS 3.0 \$36

Technical Reference Manual for the HandiPacket, 1st Rev. Ed., 2nd Printing, February 1991. The 16x clock signal is no longer routed to the 8 pin MiniDIN connector.

When changing DIP switch settings, or if HandiPacket malfunctions, be sure DIP switch 'CLICKS' fully into the desired position.

Radio cable color coding:	<u>Black 18 Inch Cable</u>	<u>Gray 60 Inch Cable</u>
Shell -	Shield	Shield
Pin 1 -	Brown	Violet
Pin 2 -	Red	Black
Pin 3 -	Orange	Yellow
Pin 4 -	Yellow	Orange
Pin 5 -	Green	Blue
Pin 6 -	Blue	Green
Pin 7 -	Violet	Brown
Pin 8 -	Black	Red

TNC-200:

Current Firmware: 32k RAM, with PMS: P1.1.6D4 PMS 3.0 \$2F
European TNC-200, 32k RAM, with PMS: E1.1.6D4 PMS 3.0 \$41

TNC-220:

Current Firmware: 32k RAM, with PMS: P1.1.6D4 PMS 3.0 \$1D
European TNC-220, 32k RAM, with PMS E1.1.6D4 PMS 3.0 \$8F

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 product is manufactured on a circuit board with the identification of DFM-MC. The parts placement on this circuit board is quite different than earlier versions. The manual still refers to the older version circuit board, but all the part and connector specifications are the same. The schematic shipped with the MC-NB96 is up to date.

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.

The Transmit Audio (TXA) coaxial cable from the MC-NB96 is labeled with a spot of gold paint or TXA label.

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 and 17 & 18 on the underside of the modem disconnect header of the packet controller. If recovered clock from the MC-NB96 is used, also cut the trace between pins 13 & 14.

On TINY-2 and MICROPOWER-2 TNCs which are several years old, it may be necessary to shorten JPL and JPD and to bend C25 to the side to clear the MC-NB96 board.

EM-NB96:

The EM-NB96 is currently undergoing revision.

PSK-1 and PSK-1T:

Current Firmware: (PSK-1) Version 2.10, 12/4/91, \$251A; (PSK-1T modem) Version 3.10, 12/2/91, \$A4C8, (PSK-1T TNC) 1.1.6D5, 12/2/91, \$2D.

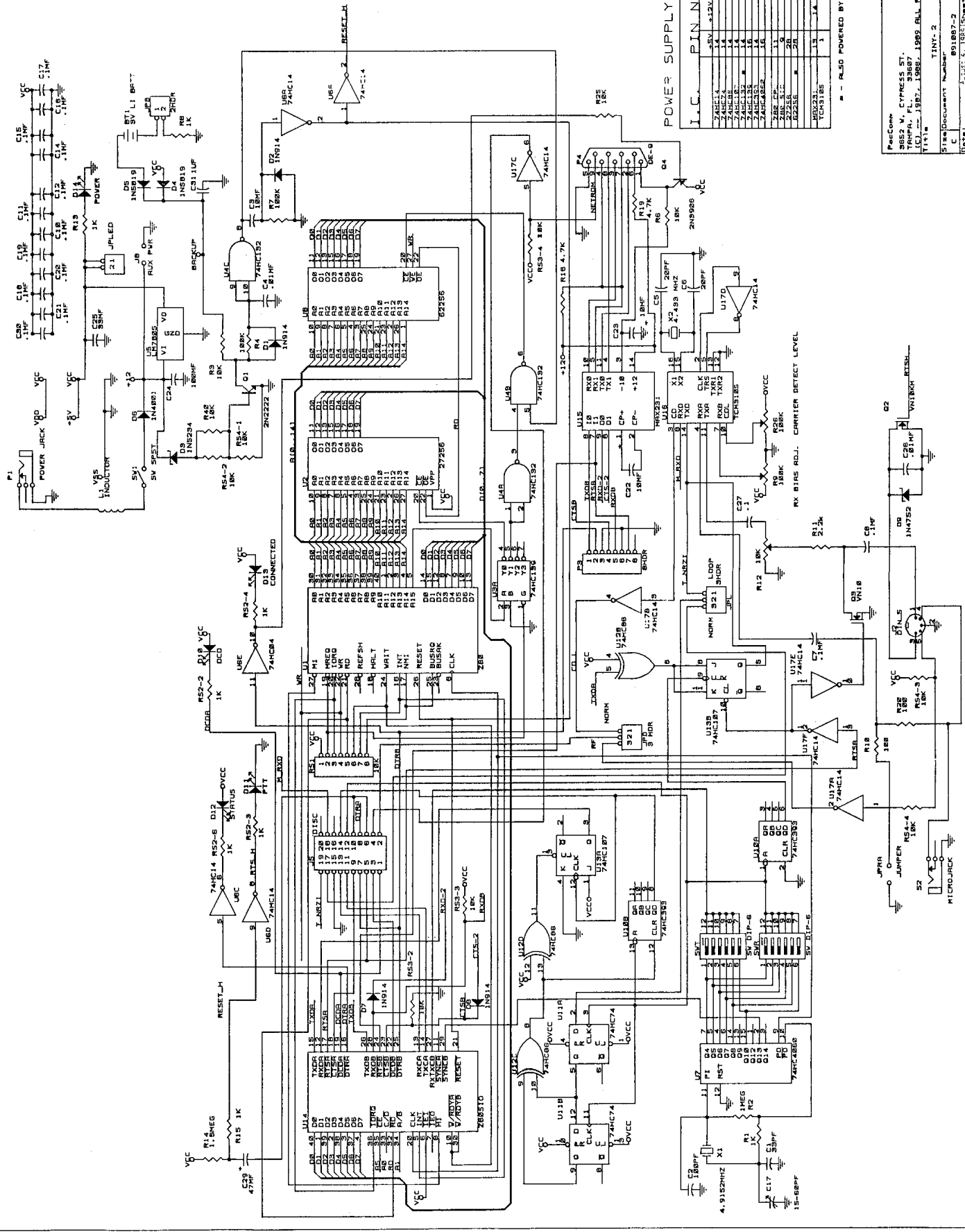
Instruction Manual, Second Edition, December 1991.

Page 31, 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.

The following production change has been made for increased PSK-1 output. The value changes may not be shown on the schematic. R41 changed to 13k and C66 changed from 0.1uf to 0.2 uf.

PSK-1 cable color coding: 8 pin DIN cable 5 pin DIN cables Alternate 8 pin DIN cable
(Confirm with Ohm meter.)

Pin 1	Red	Yellow	Red
Pin 2	Yellow	Red	Green
Pin 3	Blue	Black	Black
Pin 4	Orange	Orange	White
Pin 5	Green	Brown	Yellow
Pin 6	Brown		Brown
Pin 7	Violet		Orange
Pin 8	Black		Blue



POWER SUPPLY TABLE

IC	PIN NUMBERS
U1	1 4 5 14 15 18
U2	1 4 5 14 15 18
U3	1 4 5 14 15 18
U4	1 4 5 14 15 18
U5	1 4 5 14 15 18
U6	1 4 5 14 15 18
U7	1 4 5 14 15 18
U8	1 4 5 14 15 18
U9	1 4 5 14 15 18
U10	1 4 5 14 15 18
U11	1 4 5 14 15 18
U12	1 4 5 14 15 18
U13	1 4 5 14 15 18
U14	1 4 5 14 15 18
U15	1 4 5 14 15 18
U16	1 4 5 14 15 18
U17	1 4 5 14 15 18
U18	1 4 5 14 15 18
U19	1 4 5 14 15 18
U20	1 4 5 14 15 18
U21	1 4 5 14 15 18
U22	1 4 5 14 15 18
U23	1 4 5 14 15 18
U24	1 4 5 14 15 18
U25	1 4 5 14 15 18
U26	1 4 5 14 15 18
U27	1 4 5 14 15 18
U28	1 4 5 14 15 18
U29	1 4 5 14 15 18
U30	1 4 5 14 15 18

== ALSO POWERED BY BATTERY