Kantronics

KPC–3 Plus

Users Guide: **Introduction,**
Getting Started,
Modes of Operation,
Command Reference, and
Hardware Specifications

Contact Us:

Kantronics

14830 W. 117th St.
Olathe, Kansas 66062

Orders / Inquiries	(913) 839-1470 FAX (913) 839-8231 E-mail sales@kantronics.com Website: www.kantronics.com
Service / Technical Support	(913) 839-8143 (8 AM to 12PM and 1 to 5 PM Central Time, M-F) FAX (913) 839-8231 E-mail service@kantronics.com

REVISIONS

Revision	Date	Description
A	2003-11-11	Inserted warranty form and revisions page. Deleted reference to previously supplied molded audio plug with shielded cable.
B	2005-04-28	Added UIDUPE command definition and reference in a number of locations to the GPS port. Updated “Expanding the RAM”.
C	2005-09-21	Deleted further references to previously supplied molded audio plug with shielded cable. Deleted reference to Warranty Registration card. Changed time to file Warranty Registration from 10 to 60 days. Added e-mailing of Warranty Registration information. Other cleanup.
D	2005-09-26	Deleted reference to Kantronics HSP cable.
E	2006-05-10	Changed Kantronics address to 3115 W. 6th St., Ste. A.
F	2006-08-22	Removed CE mark pending RoHS Compliance.
G	2011-07-31	Updated contact information, minor text revisions.
H	2015-05-28	Updated to a searchable format.

The KPC-3 Plus is a Kantronics hardware and software design incorporating the AX.25 Level 2 Version 2 Packet protocol as adopted by the American Radio Relay League.

We have attempted to make this manual technically and typographically correct as of the date of the current printing. Production changes to the TNC may add errata or addendum sheets. Your comments and/or suggested corrections can be sent to us (see our contact information).

This document was compiled in the U.S.A.

The KPC-3 Plus is manufactured in the U.S.A.

© Copyright 2007-2015 by Kantronics. All Rights Reserved.

Contents of this publication or the firmware within the KPC-3 Plus may not be reproduced in any form without the written permission of the copyright owner.

KPC-3 Plus is a registered trademark of Kantronics Co., Inc.
KPC-9612 Plus is a registered trademark of Kantronics Co., Inc.
KAM XL is a registered trademark of Kantronics Co., Inc.
NET/ROM is a registered trademark of SOFTWARE 2000.
APRS is a registered trademark of Bob Bruninga, WB4APR.
HyperTerminal is a registered trademark of Microsoft.

Warranty Registration

Please take the time to (print this page) fill out the warranty registration form and mail it to Kantronics, including a copy of your sales receipt, to register your purchase. Refer to the warranty policy in this manual for further information.

Mail form and sales receipt to:

Kantronics
14830 W 117th Street
Olathe, KS 66062

Warranty Registration

Last Name:_____ First Name:_____
Call Sign:_____ .
Mailing Address:_____
City:_____
State:_____ Zip/Postal Code:_____ Country:_____
Telephone:_____ E-Mail:_____
Product: KPC-3+_____ Serial #:_____
Date of Purchase:_____ Dealer:_____

IMPORTANT

READ THIS PAGE BEFORE YOU INSTALL YOUR NEW KANTRONICS PRODUCT

This product contains SOFTWARE on Programmable Read Only Memory (PROM) and/or diskette and/or CD, which is protected by both United States copyright law and international treaty provisions.

If you install or use this product, you will be deemed to be bound by the terms of the SOFTWARE license shown below. If you do not wish to be bound by such license, return such product and all associated documentation unused to your supplier for refund of the amount you paid.

License Agreement

1. **License.** In consideration of payment of the License Fee, which is included in the price of the product, the Licensee (you) is granted by the Licensor (Kantronics Company, Inc. - Kantronics) a non-exclusive right to use the SOFTWARE and associated documentation. No ownership rights to the SOFTWARE or its Documentation are transferred from Kantronics to you.

2. **Term.** This License Agreement is effective until terminated. You may terminate this Agreement by destroying the PROM or diskette or CD and documentation. You may not rent or lease the SOFTWARE, but you may transfer the SOFTWARE and accompanying written materials on a permanent basis provided you retain no copies and the recipient agrees to the terms of this Agreement. Kantronics may terminate this Agreement without notice if you violate any terms or conditions of the Agreement. In the event of termination of the Agreement, provisions relating to Kantronics' disclaimers of warranties, limitation of liability, remedies, or damages and Kantronics' proprietary rights shall survive.

3. **Object Code.** The SOFTWARE is delivered in object code only. You shall not reverse compile or otherwise reverse engineer the SOFTWARE.

4. **Limited Warranty.** This product is covered by the standard Kantronics Limited Warranty, which is enclosed.

5. **General.** This License Agreement constitutes the complete Agreement between you and Kantronics.

The SOFTWARE and/or Documentation may not be exported or re-exported in violation of any export laws or regulations of the United States of America or any other applicable jurisdiction.

This Agreement shall be governed by and interpreted under the laws of the State of Kansas, United States of America.

Use, duplication, or disclosure by the Government of the United States is subject to restrictions as set forth in subparagraph (c)(1)(ii) of the Rights in Technical Data and Computer SOFTWARE clause of DFARS 252.227-7013.

Kantronics may in its sole discretion, provide you with upgrades of the SOFTWARE and/or Documentation if you have provided Kantronics your completed Warranty registration with a copy of your receipt showing the amount you paid.

LICENSEE ACKNOWLEDGES HAVING READ AND UNDERSTOOD THIS AGREEMENT AND AGREES TO BE BOUND BY ITS TERMS. LICENSEE FURTHER AGREES THAT THIS AGREEMENT IS THE COMPLETE AND EXCLUSIVE STATEMENT OF THE AGREEMENT BETWEEN LICENSEE AND LICENSOR AND SUPERSEDES ANY PROPOSAL OR PRIOR AGREEMENT, ORAL OR WRITTEN, AND ANY OTHER COMMUNICATIONS RELATING TO THE SUBJECT MATTER OF THIS AGREEMENT.

Any questions concerning this Agreement or any other matter relating to Kantronics, Kantronics products, or business practices, may be sent to us by any of the means on our contact information page.

TABLE OF CONTENTS

Contact Us: **2**
REVISIONS **3**
Warranty Registration **4**
License Agreement **5**
LIMITED WARRANTY **9**
Return/Repair Procedures 12
Radio Frequency Interference Statement 14
RFI Suppression 15
FCC Declaration of Conformity: 16
Introduction **17**
Welcome 17
Overview of This "User's Guide" Manual 17
Major Uses of Your KPC-3 Plus 17
Package Contents 18
Additional Parts for Your Packet Radio Station 18
Our Assumptions about You 19
Documentation Conventions 19
Overview of Packet Radio 19
.Inside a TNC – the KPC-3 Plus 26
Inside a TNC – the KPC-3 Plus 27
Installing Your KPC-3 Plus **29**
The Major Components of Your Station 30
Back Panel 30
How the Parts of Your Station are Connected 33
Connect Your KPC-3 Plus to a Power Source 34
Connect your KPC-3 Plus to Your Computer 36
Configure Your KPC-3 Plus 39
Connect the KPC-3 Plus to a Transceiver 43
Connecting to a GPS Device (Optional) 53
Getting Started **54**
The Front Panel of the KPC-3 Plus 55
Beginning a Packet Session 55
Giving Commands and Transmitting Data 56
NEWUSER Commands 57
Using NEWUSER Commands 59
Connect to Your Mailbox 60
Monitor Communications From Nearby Stations 61
Communicate Directly with a Nearby Station 62
Modes of Operation **64**
Packet Mode of Operation 64
Remote Access to Your TNC 74
PBBS (Personal Mailbox) **76**
GPS NMEA Interfacing **91**
GPS Operations 95
KA-Node **104**
Introduction to Basic Packet Networking **112**
K-Net Network node **114**
K-Net Node Commands 115
WEFAX Mode 137
Other Modes of Operation 139
Modem Mode 140
Kantronics Host Mode Operation 141
KISS Mode 142
Command Reference **146**
Introduction 146
Commands **151**

Appendix A: Advanced Installation **209**
Precautions 209
Cable Wiring 209
Connecting to the Computer (DSUB-25) 209
Connecting to your Radios 212
Appendix B: Advanced Information **215**
Assembly and Disassembly 215
Hard Reset 215
Calibration/Equalization 216
PTT (Push-to-Talk) Watchdog Timer 217
Microprocessor Watchdog Timer 217
A/D Converter 217
KPC-3 Plus Jumpers 218
Appendix C: Options for the KPC-3 Plus **221**
Low Power Operation 221
Expanding the RAM in the KPC-3 Plus 221
Installing the Optional Real-time Clock Module 221
Replacing the Lithium Battery 222
Appendix D: In Case of Difficulty **223**
KPC-3 Plus Does Not "Sign-On" to Computer 223
You Are Unable to Make a "Connect" 223
Cannot Transmit 223
Cannot Return to Command Mode 224
Appendix E: Additional Information **225**
Specifications 225
Messages from the KPC-3 Plus 226
ASCII Chart 232
KPC-3 Plus Parts List 233
KPC-3 Plus Parts List 233
KPC-3 Plus MX Parts List 234
KPC-3 Plus Parts Layout 235
KPC-3 Plus MX Parts Layout 236

LIMITED WARRANTY

KANTRONICS CO., INC.
LIMITED WARRANTY
Effective January 1, 1997

To receive notice of future updates, new product information and prompt warranty service, please fill in the Kantronics Warranty Registration form **COMPLETELY** and return it along with a copy of proof of purchase (to establish purchase date) by any means to us (see our Contact Us page).Warranty Registration form and proof of purchase may be e-mailed to sales@kantronics.com.

NOTE: Return of the Warranty Registration form and proof of purchase is a pre-condition to warranty coverage.

1. **WARRANTY.** Kantronics warrants to the first consumer purchaser ("you"), for the Applicable Warranty Period (as described below), that the Applicable Product (as described below) will be free from defects in material and workmanship.

2. **REMEDY.** Kantronics agrees that, for any Applicable Product found by Kantronics to be in violation of the warranty of Section 1 hereof within the Applicable Warranty Period, it will, at its option, repair or replace the defective Applicable Product at no charge to you, excluding in-bound shipping charges.

3. **EXCLUSIVE REMEDY.** Repair or replacement of the Applicable Product, as provided herein, is the sole remedy available to you against Kantronics, and in no event will Kantronics be responsible for any other liability or damages or for incidental, special, or consequential damages, regardless of whether purported liability is predicated upon negligence, strict tort, contract, or other products liability theory and whether or not Kantronics is warned about the possibility of such liability or damages. **SOME STATES DO NOT ALLOW THE EXCLUSION OR LIMITATION OF INCIDENTAL OR CONSEQUENTIAL DAMAGES, SO THE ABOVE LIMITATION OR EXCLUSION MAY NOT APPLY TO YOU.**

4. **DISCLAIMER.** This Limited Warranty is in lieu of all other warranties expressed or implied and no representative or person is authorized to assume for Kantronics any other liability in connection with the sale of its products. KANTRONICS SPECIFICALLY DISCLAIMS THE IMPLIED WARRANTY OF MERCHANTABILITY AND IMPLIED WARRANTY OF FITNESS FOR A PARTICULAR PURPOSE FOR ANY APPLICABLE PRODUCT. IF, HOWEVER, YOU ARE A CONSUMER WITHIN THE MEANING OF 15 U.S.C. 2301(3), THE ABOVE DISCLAIMER OF IMPLIED WARRANTIES IS EFFECTIVE ONLY FOR PERIODS OUTSIDE THE APPLICABLE WARRANTY PERIOD. **SOME STATES DO NOT ALLOW LIMITATIONS ON HOW LONG AN IMPLIED WARRANTY LASTS, SO THE ABOVE LIMITATION MAY NOT APPLY TO YOU.**

5. **APPLICABLE PRODUCTS AND PERIODS.** Kantronics products are of two types - (1) hardware units and (2) firmware and software for operation of these units, whether incorporated into the units themselves or separate from the units as

adjuncts or accessories to the units. Hardware units and the media containing firmware, software and documentation are sold to the consumer purchaser and become property of the purchaser. Firmware and software are licensed for use by the consumer purchaser in return for a fee included in the purchase price of the units and do not become the property of the consumer. (See separate License Agreement provided with these products). The products to which the warranty of Section 1 hereof applies (herein “Applicable Products”) and the periods during which the warranty shall apply to such products (herein, “Applicable Warranty Period”) are as follows:

Applicable Products:

UNITS:

KPC-3 Plus, KPC-9612 Plus, KAM XL, MT1200, MT1200G

Applicable Warranty Period: One (1) year from date of purchase.

MEDIA:

EPROMS, CDs, manuals (however bound), specification and other supplemental pages or any other media on which firmware, software or documentation are supplied

Applicable Warranty Period: Thirty (30) days from date of purchase.

6. **EXCLUSIONS.** This Limited Warranty does not apply to the cosmetic appearance of the Applicable Product; to broken or cracked cabinets; to any accessory not supplied by Kantronics which is used with the Applicable Product; to any product that has been subject to misuse abuse or overvoltage; to any product that has been modified by non-Kantronics personnel unless specifically authorized **in writing** by Kantronics; or to any product damaged or impaired by shipping (whether or not caused by poor packaging), neglect, accident, wiring not installed by Kantronics, improper parameter settings which are cleared by performing a hard reset, or use in violation of instructions furnished by Kantronics or of generally accepted industry practice. Kantronics does not warrant that the functions contained in any software will meet your requirements or achieve your intended results; or that operation of any software will be uninterrupted or error-free or without effect upon other software used with it. *Responsibility for the selection of the hardware and software program to achieve your intended results rests with you.*

7. **REMEDY PROCEDURE.** Should you need to make a warranty claim, first contact the dealer from whom you purchased the product. If the dealer is unable to assist you, contact us prior to returning an Applicable Product to receive a Return Authorization Number. (As a practical matter, problems can often be solved in such a manner without the product having to be returned to Kantronics for repair or replacement.)

Return of any Applicable Product for the enforcement of rights under this Limited Warranty shall be at your expense. Any product returned for warranty service, which Kantronics determines to be without defect or not covered by this Limited Warranty shall be subject to the minimum charge for labor and the product will be returned to you at your sole expense. Please note, no warranty service will be provided until Kantronics has been furnished with your Warranty Registration card and copy of proof of purchase establishing purchase date.

8. **NON-ASSIGNMENT.** This Limited Warranty is not assignable by you. Any attempt to assign or transfer any of the rights, duties, or obligations hereof is void.

9. **OTHER RIGHTS. This Limited Warranty gives you specific legal rights and you may also have other rights, which vary from jurisdiction to jurisdiction.**

Return/Repair Procedures

Important: Our repair statistics show that over 70 percent of the units returned for service do not, in fact, require any service. Therefore, we advise you to please double-check the following list of common, user-solvable, sources of difficulty before contacting Kantronics about returning your unit for service.

Check-List for Possible Problems

Should you encounter difficulty in getting your equipment to “talk” to your computer, please perform at least the following limited checks before calling or writing:

Carefully check your wiring connections to the 232 port.

If you purchased third-party cables, double-check to be sure that they conform to the Kantronics’ wiring instructions in this manual.

Verify your terminal baud.

It may be useful to perform a “Hard Reset”. (See Hard Reset section.) If service or repairs still appear necessary after you have checked the items listed above, it may be wise to call, fax, e-mail or write Kantronics to determine if the problem can be solved without returning the unit.

Return Procedures

When calling, report the product name and ask for the Service Department. Please have the following information available:

The unit name and serial number (the serial number is found on the bottom of the unit).

The firmware version number (the version number is displayed when you give the Version command).

If possible, you should have the unit and your computer available to perform troubleshooting operations when you call.

The Service Department telephone hours are 8:00 AM to 12:00 Noon and 1:00 PM to 5:00 PM Central Time, Monday through Friday. If you call outside these hours, the phone will just ring. The service department telephone is not connected to the main switchboard and the switchboard receptionist cannot transfer you to the service number. If lines are busy, you may wish to (and it may be faster to) contact service by fax, or e-mail. Service e-mail is checked twice per day. Before contacting us, please take the time to list out your problem fully and carefully.

When writing, faxing, or e-mailing Kantronics, include a clear description of the problem, unit name, firmware version, computer type, computer software used and if possible a list of current parameter values for your unit (as shown in a DISPLAY listing). Be sure to include a return fax number and/or e-mail address.

Returns to the factory for refund or exchange are strictly regulated. Any return for refund or exchange, must be approved by the service department.

Charges

Consult the limited warranty policy in this manual for the service provisions offered by Kantronics at no charge. This warranty is considered to be in force only when the customer has submitted his completed warranty registration within 10 days of purchase, and when the stipulations of the warranty have been met.

Violations of warranty clauses will automatically void the warranty and service or repairs will be charged to the owner.

Service outside the warranty will be charged at the cost of parts, labor, and return shipping. Units returned for service without a Return Authorization number will be subject to a minimum charge of ½ h labor plus shipping and handling.

If payment has not been previously arranged, repaired (or un-repairable) units may be returned via C.O.D.

These C.O.D. charges can be avoided by including your VISA or MasterCard number with your unit to be repaired. Shipping and repair may then be charged.

International Returns

♦ This section applies to international returns only, not to domestic returns.

In case of unit problems, first contact the dealer from whom you purchased the product. If you must return a Kantronics product to us, please observe the steps outlined below. It will save you, the customer, and Kantronics unnecessary difficulties and expense.

- All returns must be shipped to the factory.

- All expenses of returning items to Kantronics must be paid by you, including any duty/entry fees, whether the return is for warranty or non-warranty repair.

- Usually, the best way to return items to us is by mail. However, if you wish to use one of the courier services such as DHL, UPS Expedited, Federal Express, etc., be sure to use **DOOR-TO-DOOR** service. If you use one of these services, a commercial invoice may be required. Please check with your carrier before shipping.

- Include in the description of the items on the paperwork (whether postal or courier) the words:

  “U.S. GOODS RETURNED FOR REPAIR/REPLACEMENT.”

Step 1. An additional description of “Amateur radio peripheral equipment”, or “Data communications equipment”, would be helpful. It would also be helpful (but not required) to include the code number 9801.00.1035 which tells U.S. Customs agents that the package contains “U.S. goods returned without improvement/enhancement”. However, if the words “U.S. goods returned for repair/replacement” are on the paperwork, the number is not really necessary.

- Provide a value for customs purposes. This is usually the value of the item(s) in their current condition. A $0 value is not acceptable for U.S. Customs.

- Inside the package, with the item(s), include

  - a fax number and/or e-mail address (if available) in case we need to contact you

  - a correct and full address for return

  - method of payment to be used for any charges (if MasterCard or VISA, include expiration date)

  - a brief description of the problem

  - a reference to any conversations with the technical/sales staff about the problem

  - and the Return Authorization number assigned

- For warranty repairs, we will pay the shipping charges to return the item(s) to you via air parcel post. If you wish return by courier service, include your account number. To be eligible for repair under warranty, we must have a record that you sent your Warranty Registration and proof of purchase to Kantronics, and the item(s) must still be within the warranty period at the time the return is authorized.

- For non-warranty repairs, you must pay the return shipping charges.

Radio Frequency Interference Statement

Note 1: This equipment has been tested and found to comply with the limits for a Class B digital Device, pursuant to Part 15 of the FCC Rules. These limits are designed to provide reasonable protection against harmful interference in a residential installation. This equipment generates, uses, and can radiate radio frequency energy and, if not installed and used in accordance with the instructions, may cause harmful interference to radio communications.

There is no guarantee that the interference will not occur in a particular installation. If this equipment does cause harmful 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 or relocate the receiving antenna.

- Increase the separation between the equipment and receiver.

- Connect the equipment into an outlet on a circuit different from that to which the receiver is connected.

- Consult the dealer or an experienced Radio/TV technician for help.

The user is cautioned that any changes or modifications not expressly approved by the party responsible for compliance could void the user’s authority to operate the equipment. The user

is also cautioned that any peripheral device installed with this equipment must be connected with a high-quality shielded cable to insure compliance with FCC limits.

Note 2: The shield of the cable, whether foil, braid, braid over foil, or double braid, must be properly terminated (connected) 360° to the connector. This is usually accomplished by the use of a metal or metalized plastic back shell, but may be implemented by direct contact, including soldering, with metal portion of connector. Experience has indicated that cable assemblies (with connectors) advertised as “shielded” are not necessarily terminated properly, if terminated at all. Check cable construction to be sure.

RFI Suppression

In moving to the world of digital communications via computers, a new dimension of RFI may be encountered. In spite of the equipment manufacturers’ diligence, each new piece of electronic equipment will react differently in each separate environment. Every amateur station will have its own unique layout, equipment variation, and antenna installations. Experience has shown that these differences are related to the total RF environment, and may be causative factors in RFI induced problems. The suggestions given here may assist in resolving RFI problems you may encounter in your “unique” station.

- Use shielded cable for all connections between equipment.
- Make all interconnecting cables as short as practical. A balance should be maintained between cable length and equipment proximity. At times simply moving the video monitor one foot further from an interface or other device will solve a “screen hash” problem.
- Antenna runs should be kept away from equipment control lines and/or interconnecting cables. If it is necessary for such lines to cross each other they should do so at 90-degree angles.
- Ground leads should be as short as possible and go to a GOOD EARTH GROUND.
- Interconnecting cables appearing to act as radiators or antennas should be looped through a toroid. Be certain toroids, if used, are designed for the frequency in use.

FCC Declaration of Conformity:

NOTE: This equipment, Kantronics' KPC–3 Plus, has been tested and found to comply with the essential emission and immunity requirements of the EMC Directive FCC Title 47, Part 15, Subpart B. The test results are on file at the corporate offices of Kantronics.

Type of Equipment: **Information Technology Equipment**
Class of Equipment: **Class B**

CE Marking Considerations

The following cautions pertain to CE Marking of this product:

- All cables connecting to DC IN, PORT 1 (VHF), and COMPUTER must be $<$ 3 m in length
- A cable for GPS or telemetry (control output or A/D input) connected to the COMPUTER port may be $\geq$ 3 m in length.
- All cables, except for the DC IN port, must be shielded with the shield properly terminated 360° to the connector. See note 2 in the RF Interference section
- The nominal 12 V dc power must be supplied from a CE marked or third party approved power brick (wall wart) or ac to dc power supply. If a homemade power supply is used the components making up the supply must meet IEC/EN standards for such components.

Introduction

Welcome

Welcome to the Kantronics KPC-3 Plus, your pathway to amateur radio packet communication.

Please review this chapter before you install your KPC-3 Plus as part of your packet radio station.

Overview of This "User's Guide" Manual

This user's guide provides documentation on the KPC-3 Plus and packet radio, including the following:

- What equipment you will need for your packet radio station.
- A brief introduction to packet radio.
- Installing and configuring your KPC-3 Plus packet radio station and making your first connections.
- Getting started using your KPC-3 Plus.
- Documentation for each mode of operation of your KPC-3 Plus.
- A full "Command Reference", documenting all KPC-3 Plus commands.
- Full details on KPC-3 Plus jumpers, a full parts list, and other technical specifications.

Additional documentation and supporting material is available on the "Information and Program" CD that ships with the KPC-3 Plus.

Major Uses of Your KPC-3 Plus

By adding the KPC-3 Plus and a computer to your ham radio station, you can send and receive packets of digital information. This allows you to do the following:

- use computers to carry on real-time digital conversations between stations
- send, receive, store and forward mail using a personal mailbox inside the KPC-3 Plus
- send and receive mail using a community bulletin board
- send and receive files
- get and re-transmit location data from Global Positioning System devices

- serve as a digipeater for other stations
- serve as a network node point for other stations, using KA-NODE and/or the optional K-Net feature

And much more.

Package Contents

Check to see that you have the items listed below (later you will see how these items are used in your system):

- ♦ KPC-3 Plus unit
- ♦ parts to use in assembling cabling
  - Male DSUB-9 connector for radio port
  - Metalized DSUB-9 back shell with hardware
  - 3 foot (0.91 m) piece of 5-conductor shielded cable to connect the KPC-3 Plus to your radio
  - 2.1 mm dc power connector
- ♦ and, of course, this “User’s Guide” manual on a CD-ROM

Additional Parts for Your Packet Radio Station

In addition to your KPC-3 Plus unit, you will need the following parts to set up your packet radio station:

- ♦ An FM transceiver
- ♦ A microphone (Mic) plug and/or mating accessory plug or external audio plug for your radio
- ♦ A computer with an RS232 serial port (or other RS232 terminal device)
- ♦ A serial modem cable, used to connect the KPC-3 Plus to your computer
- ♦ A 12 V dc power supply or power adapter
- ♦ (optionally) 9 V battery and battery clip (not supplied, you must install)

Our Assumptions about You

We assume that you are familiar with the following or that you can get help on these topics if necessary:

- general familiarity with your radio equipment and its intended uses
- basic use of your computer and its operating system, including copying disks and files, working with directories, and identifying and using the serial (COM) ports on your computer
- basic electronics needed if you are going to build or upgrade hardware yourself, (e.g., making cables)

Documentation Conventions

The following conventions are used in the KPC-3 Plus documentation:

To indicate a particular key, the name of the key is given in capitals. For example, press the ENTER key.

Sometimes you need to hold down one key on the computer while pressing another key. This is indicated by giving the name of the first key, then a plus '+', then the name of the second key. For example, "Ctrl+C" means "press the key labeled 'CTRL' or 'Ctrl' (i.e., the "control" key) and, while continuing to hold it down, press the 'C' key". Multiple-key combinations that generate a single character are shown in angle brackets, like this: <Ctrl+C>.

Conventions for the KPC-3 Plus commands are covered in the "Command Reference" section of this manual.

Overview of Packet Radio

This section gives a brief overview of packet radio, for those who are new to packet radio and those who want to review the topic.

Topics covered are:

- The three basic components of a packet radio station
- Sending a message from one station to another
- Details about packets
- A brief discussion of protocols
- An overview of the insides of the KPC-3 Plus

If you are already familiar with packet radio, you may wish to skip the rest of this chapter, or give it a quick review.

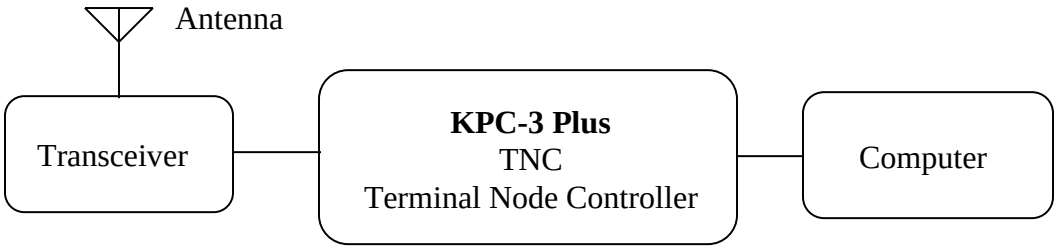
Packet radio gives you the ability to send essentially error-free digital communications to other packet radio stations. As a sender or receiver, you see just the messages, or files, being sent and received. But there is much going on beneath the surface. You will get more out of packet radio, and have an easier time getting started, if you have at least a basic understanding of what is involved in packet radio communication.

+ Do not be concerned here with ***how*** to do things —that comes later.

Three Basic Components of a Packet Radio Station

A packet radio station has three basic parts:

- a transceiver, with an antenna,
- a device called a TNC (i.e., Terminal Node Controller), which is a combination modem and special-purpose micro-computer, and
- a general purpose computer (or a terminal).



The three parts of a packet radio station work together as follows:

+ The **transceiver:** (1) sends and receives radio signals to and from your antenna and (2) passes audio signals back and forth between itself and the TNC.

+ The **TNC** (Terminal Node Controller): (1) translates audio signals into digital information and vice versa, (2) performs a number of control and information storage functions, and (3) communicates digitally with your computer.

+ The **computer** communicates digitally with the TNC, so you can: (1) view messages received from the transceiver or stored in a mailbox (i.e., PBBS), (2) use the computer to send data to, and receive data from, other stations, via the TNC and your transceiver, and (3) control the operation of the TNC.

Sending a Message to Another Station

A quick way to see the basic components of packet radio in action is to follow a simple message as it goes from one station to another, via an intermediate station.

Let’s assume that an originating station, WØXI, wants to send the message “HELLO” to a destination station, KBØNYK. Also, assume that WØXI knows that his signal might not

reach KBØNYK's station directly, but that it can be picked up and repeated by an intermediate station, NØGRG, and then received by KBØNYK.

The following are the basic steps of a station-to-station packet communication, as seen from the point of view of the message, "HELLO," sent from source station WØXI to destination station KBØNYK, through intermediate station NØGRG. Step 1 is one-time configuration task, Steps 2-3 establish a connection between stations, and Steps 4-13 follow the "HELLO" message from the time it is created by WØXI to when it is read by KBØNYK.

Step 1. Before this message can be transmitted from a source station to a destination station, via an intermediate station, all the stations involved have to have **callsigns**, which are used by the TNCs involved to identify and process information (e.g., source, destination, and routing of information).

So we start by assuming each operator involved in this example has already performed this one-time configuration task, creating the callsigns given above. Stations are identified by reference to their callsigns. Note that a callsign can be for a TNC (e.g., KPC-3 Plus) or for a Personal Bulletin Board System (PBBS) "mailbox" that resides in RAM inside the TNC.

Step 2. This communication begins when the operator of station WØXI turns on his computer, TNC (e.g., a KPC-3 Plus), and transceiver of his station and uses a computer software program to tell the TNC that he wants to establish a line of communication, called a "connection," with a "destination" station KBØNYK using an intermediate station NØGRG as a repeater station to carry the information beyond the range of the originating station and into the range of the destination station.

Step 3. The "connect" command with address given to the TNC causes the TNC to formulate a "request to connect" message that is sent from WØXI. If this message is detected by the specified intermediate station (i.e., NØGRG), repeated, and then detected by the destination station (i.e., KBØNYK), a connection may be established.

To confirm a connection, the destination station automatically sends a special message back though the reverse path, to the originating TNC.

When this "confirmation of connection" is processed by WØXI's TNC, the TNC assumes that the operator wants to begin sending messages. So the TNC: (1) puts a message on the computer monitor confirming that WØXI is now connected, through intermediate station NØGRG, to KBØNYK, (2) turns ON the "connected" indicator on the TNC, which stays ON as long as the connection is in effect, and (3) switches itself from taking commands from the computer (i.e., TNC in "command mode") to serving as a message-processor, sending and receiving messages to and from the connected stations (i.e., TNC in "conversation mode").

Step 4. Now that WØXI is in “conversation” mode, the operator types in the message, “HELLO”, and presses the ENTER key on the computer to indicate the end of the message and the beginning of the message’s journey.

Step 5. The computer sends the message, “HELLO,” to the TNC for processing.

Step 6. The TNC receives the message, “HELLO”, and takes charge of the situation, performing a number of complex operations: it uses the “connection” information about where the message is to go and the path it is to take and combines this with the message itself to assemble a “packet” of digital information for transmission from the transceiver.

The details of what is in the packet and how it is organized are fully defined by agreed-upon standards, so any machine that knows the “rules” can make use of the information.

The standard for amateur packet radio at this time is AX.25, so let’s assume the packet assembled in the TNC is an AX.25 packet.

Technical note: *All amateur radio packets use the AX.25 standard, but it is possible to use another standard or protocol (e.g., TCP/IP) “on top” of AX.25, in which case the packet is referred to by the name of the highest-level protocol.*

Step 7. After the TNC assembles the “HELLO” message into a packet of digital information in the TNC, it passes the packet on to a modem, which is also in the TNC, which transforms the information into a series of audio signals for serial transmission to station WØXI’s transceiver.

Step 8. The TNC orders the transceiver to send a message (i.e., keys push-to-talk) and then sends the audio version of the “packet” through the radio port of the TNC, up a cable to the microphone connector on station WØXI’s transceiver, and to the transmitter.

Step 9. The packet is transmitted by WØXI’s transceiver as a burst of radio-frequency signals.

Step 10. The “HELLO” packet, including address information (i.e., to, from, via) and other control information, is now on the air, where it may be correctly interpreted by all appropriately configured (i.e., AX.25 packet) stations that receive the signal.

Step 11. KBØNYK is on the air and is running AX.25 packet, but it is too far away from WØXI to pick up the signal, so does not get the message as it is sent from WØXI.

Step 12. As we know from the “connection,” already established, intermediate station NØGRG, an AX.25 packet radio station, is on the air, and is close enough to receive the radio burst sent by WØXI. It receives the burst of radio signals from WØXI and passes the audio signals on to its TNC for processing. The TNC (e.g., a KPC-3 Plus) transforms the signal from audio to digital, analyzes

the series of 1's and 0's in the incoming bit-stream, and identifies the burst as an AX.25 packet of information containing the message, "HELLO."

Also, and most significantly, NØGRG registers that the packet designates NØGRG as the first (and only) intermediate station on a path between a source, WØXI, and a destination, KBØNYK. As a fully functioning AX.25 packet station, NØGRG knows what to do: it sends the message back out over the air (i.e., digipeats it), where it may or may not be received by its intended destination, KBØNYK. Before doing so, it marked its call in the packet to indicate to others that it had seen the packet and had sent it on its way.

Note: Since a "connection" has been established, we expect the message to get through, but each packet is processed separately, and each must make it on its own.

Step 13. As we know from the "connection," already established, KBØNYK, an AX.25 packet radio station, is on the air, close enough to receive the RF burst sent by NØGRG, and currently set to display messages on a computer screen. So the "HELLO" message re-transmitted by NØGRG arrives at KBØNYK's TNC (e.g., a KPC-3 Plus), where it is processed and recognized as an AX.25 packet addressed to itself. Then KBØNYK's TNC sends the message, "HELLO," to the computer screen for KBØNYK to read. At the same time, KBØNYK's TNC creates an acknowledgment packet to send back to WØXI's TNC, reversing the address path it found in the "HELLO" packet. When KBØNYK's TNC detects that the radio channel is clear, it keys the transmitter and sends the packet back over the air.

Step 14. NØGRG's TNC now recognizes another packet which has its callsign in the address path, requesting it to be digipeated. NØGRG's TNC does so as soon as it can.

Step 15. WØXI's TNC now hears a packet addressed to it. Since the digipeated packet originally came from KBØNYK, to whom he is connected, and since the packet contains an acknowledgment for data that he had recently sent, WØXI's TNC can rest easy knowing that the data he sent was properly received.

This concludes our close-up look at a single packet communication event, showing how the various parts of packet radio work with each other. To simplify and focus this example, various details and possible complications were ignored, but this still shows the basic steps involved in establishing a connection and sending the message "HELLO" from one person/station to another person/station.

This example covers one of many uses of packet radio. As covered in other sections of this manual, there are many ways to communicate with other stations.

Now that you have followed one particular message from one station to another, it is useful to learn about the "packets" in packet radio.

Packets: Dividing Messages into Segments

An efficient way to send a message from one place to another is to break the information into small segments, called packets or frames, and send each packet separately.

Note: The correct technical term is “data frame”, not “packet” or “frame,” but we will use the term “packet,” since that is familiar to most people and is common in the field.

Your packet radio station automatically divides your communication (e.g., a message or file) into small segments (see the PACLEN command for details) and puts each segment into its own “packet”. Each packet includes information used for addressing and error correction.

Unconnected Packets

Packets do not need to be addressed to any particular station. Packets transmitted without any established connection between sender and receiver are “unconnected” packets. “Unconnected” packets are just sent out without any back-and-forth coordination with any receiving station. Stations receiving these packets can check to see if the packets are correct (i.e., the same bit pattern is received as when they were sent), but they cannot request any retransmission of faulty packets. Nor can they request re-transmission of any packets in a series that are not received at all. This is because “unconnected” packets do not include sequence information that could be used to track them in order and also because a TNC sending or receiving “unconnected” packets is not set up to do any communicating about the success or failure of its packet transmissions.

+ Note: Packets sent as “unconnected” are also called UNPROTO (non-protocol), because the sender expects no response from the receiving station.

Connected Packets

Packet radio communication is often done between two stations that are connected together. “Connected” communication using packet radio assures transmission with virtually 100% accuracy by having the sender and receiver follow a set of rules for communicating. These rules are specified in the AX.25 protocol.

The following is an overview of what happens during a packet radio communication between two connected stations.

After a connection has been established between two stations, information Packets are transmitted one-by-one (or in small batches), with a sequence number as part of each packet, from the sending station to the receiving station. As the communication proceeds, the receiving station keeps track of which packets it has received error-free (as determined by using the check-sum that is part of each packet) and uses this to tell the sending station what packet(s) to send next.

Using these rules of communication, a packet radio connection between a sender and a receiver provides virtually error-free transmission of information, until the

communication is ended by the parties, or until the sender “times out” after trying a number of times to get feedback and not succeeding.

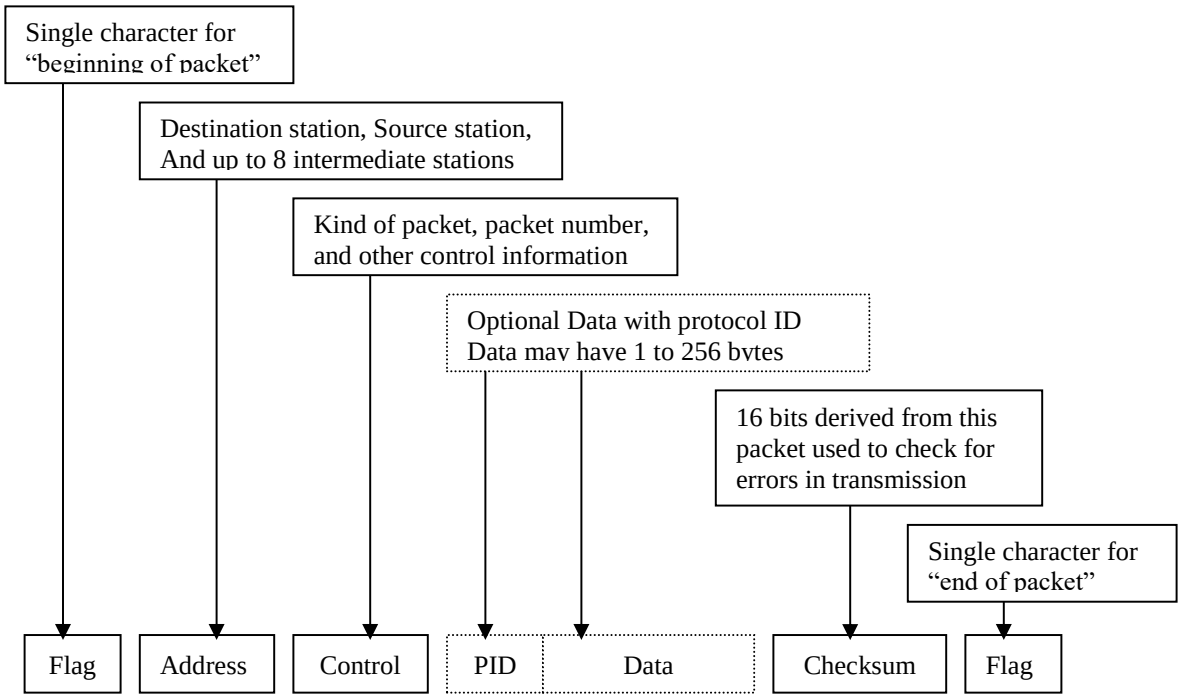
+ **Hint:** In most cases, communication by packet radio is like mail, not the telephone, in that it is not done in real-time, but via messages that are routed from one station, or “node,” to another.

How a Packet is Organized

Your packet radio station takes care of all the details of working with packets, so you do not need to see the “inside” of packets. But it is useful to have at least a basic idea of how they are organized.

All amateur radio packets (also called “frames”) are defined by the AX.25 protocol, which is discussed in the next section. Packets used to carry messages, or chunks of messages, as in the “HELLO” example, are called “information packets.”

The following diagram shows the basic building blocks of “connected information” packets used in amateur packet radio:



Kinds of Packets

As shown in the above diagram, the “control” field in an AX.25 packet includes a code telling what kind of packet the current packet is. In addition to connected information packets, the AX.25 protocol also defines two other kinds of packets, each of which is nearly identical in structure to the connected information packet: control packets (of

which there are several kinds) and unnumbered packets. To learn more about these, see the documentation for the KPC-3 Plus' MCOM command.

As noted at the beginning of this section, the organization of amateur radio packets is defined by the AX.25 protocol. Now it is useful to give a brief discussion of this protocol, and another that fits "within" it.

Protocols: Rules for Working Together

Packet radio, like any communication, is only possible if all parties involved (i.e., sender, receiver, and intermediate points on the route between them) agree on the rules for organizing information and the rules for dealing with all steps of the transmission. These rules are called "protocols," or standards.

AX.25

The defacto standard protocol for amateur packet radio communication is AX.25 (level 2, version 2). For details on AX.25, see the ARRL publication, **AX.25 Amateur Packet-Radio Link-Layer Protocol.**

┼ **Hint:** You can set the KPC-3 Plus to use an earlier protocol, AX.25 (level 2, version 1)

Alternatives to AX.25

TCP/IP and the KISS protocol: The TCP/IP (Transmission Control Protocol/ Internet Protocol) suite of protocols require functionality not found in native AX.25 packets, so Phil Karn KA9Q defined a protocol called KISS that implements TCP/IP functionality for amateur radio. For further information on KISS see the KISS Mode section of the "Modes of Operation" chapter.

XKISS is an extension of the KISS protocol, defined by John Wiseman G8BPQ.

For further information on XKISS, see the XKISS Mode section of the "Modes of Operation" chapter

▪

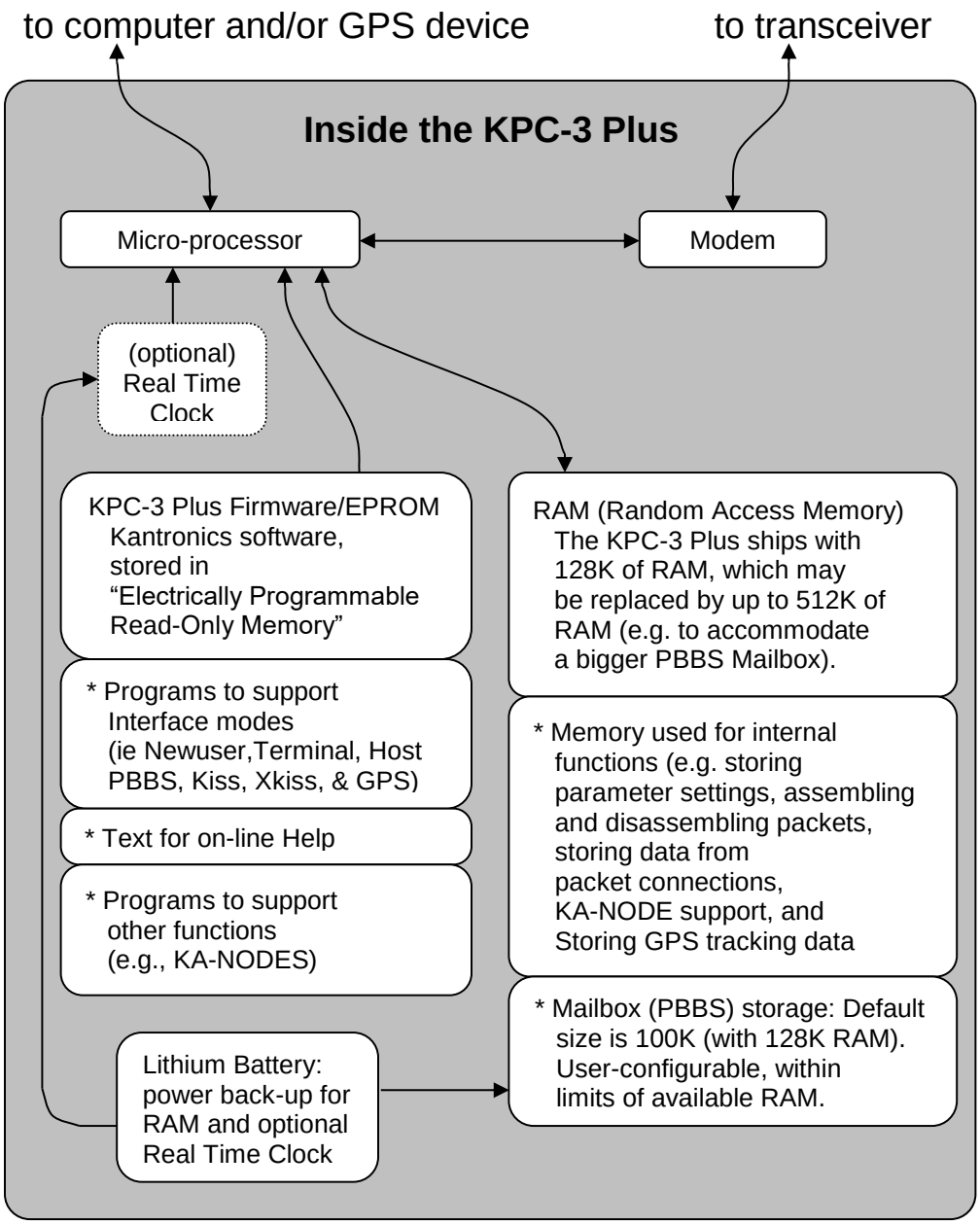
Inside a TNC – the KPC-3 Plus

To better understand the workings of your packet radio station, it is helpful to have an overview of the major components of the TNC device at the center of your station.

In the early days of packet radio, TNC's consisted primarily of a modem for communicating with a transceiver and a special-purpose microprocessor called a PAD (i.e., packet assembler/disassembler) used to process packets of information and communicates with a computer or terminal.

+ As packet radio has developed, more and more features have been added, such as memory and software dedicated to a "Personal Bulletin Board System" within the TNC. The following diagram shows the most important internal components of a TNC (the KPC-3 Plus). The diagram is not to scale. For a detailed example, see the parts list and layout diagram for the KPC-3 Plus in the appendices of this manual.

Note: Other TNC's may have different features than the KPC-3 Plus for example, the ability to operate two radios at once.



This concludes the introduction to packet radio. The next chapter guides you through the steps of installing your KPC-3 Plus.

Installing Your KPC-3 Plus

The goal of this chapter is to guide you as quickly as possible through the steps of setting up your packet radio station.

The major steps involved in installing (i.e., connecting the parts) your packet radio station are as follows:

- connect your KPC-3 Plus to a power source and confirm that the unit powers on when it should,
- connect the “Computer” port on the KPC-3 Plus to a serial port on your computer, install terminal communication software in the computer if necessary, and configure the terminal software and the KPC-3 Plus so digital information can be sent back and forth between the KPC-3 Plus and the serial port on the computer,
- connect the “Radio” port on your KPC-3 Plus to your transceiver and adjust the volume of the transceiver, so the KPC-3 Plus can receive signals from the transceiver for processing and the KPC-3 Plus can send signals to the transceiver for transmission,
- (optionally) use the “Computer” port on the KPC-3 Plus to connect your KPC-3 Plus to a GPS device, so the KPC-3 Plus and the GPS device can send digital information back and forth.

To show you where you are heading with the step-by-step configuration covered below, we start with a brief section on equipment needs for your station and an overview diagram that shows how the parts of your station will be connected when you are ready to begin using it.

The Major Components of Your Station

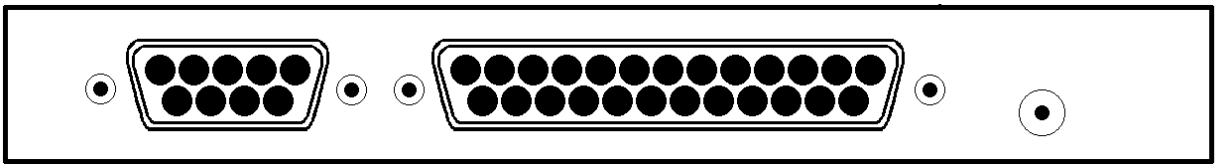
This section describes the basic components needed for your packet radio station and the connections between components that will be involved in the installation of your station.

The KPC-3 Plus

Your KPC-3 Plus is going to be at the center of your packet radio station, so setting up your station involves connecting other units to your KPC-3 Plus.

As shown below, the back of the unit has connectors to connect the KPC-3 Plus to your transceiver, your computer (or a GPS device), and a power source:

Back Panel



Radio Port
DSUB-9 female

Computer Port
DSUB-25 female

Power Jack
2.1 mm

Radio Port	This DSUB-9 female connector accepts the cable from your radio, for 1200 baud packet operation.
Computer	This DSUB-25 female connector accepts the cable from your computer serial port. The KPC-3 Plus communicates with RS232 levels using standard ASCII characters for all commands.
Power jack (2.1 mm)	This connector is provided to apply external power (6 to 25 V dc) to the KPC-3 Plus. The center post is the positive connection and the sleeve is negative (ground).

Note: All Kantronics' TNC's can operate without the computer being connected, once they have been configured. For example, you can receive and store messages in the

personal mailbox inside the KPC-3 Plus without using your computer. And your KPC-3 Plus can serve as a relay station for other stations without your computer being connected. This independence from the computer is possible because the TNC contains the intelligence necessary to carry out these functions, once it is appropriately configured and attached to a transceiver.

The Transceiver

Your KPC-3 Plus will operate with most FM transceivers.

There are currently hundreds of models of FM transceivers. **You will need to refer to the documentation for your transceiver to determine how it must be connected to external devices, such as the KPC-3 Plus.**

Specifically, you will need to know:

- what type of microphone connector to use with your transceiver, and
- which pins connect to which in the cabling between the KPC-3 Plus and your transceiver's microphone

Wiring diagrams for some common transceivers are included later in this chapter for your assistance.

The Computer

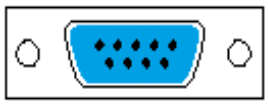
Your KPC-3 Plus can work with any computer that can communicate with it through a standard RS232 serial modem port.

Technical note*: You need to be sure the wiring is as shown in the installation section below. You may, of course, make your own cable; make sure it is correctly wired and shielded.*

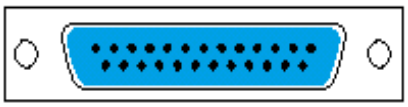
The Serial Port on Your Computer

You will be using a standard (RS232) modem cable (or making a cable with the same wiring) to connect your KPC-3 Plus to a serial (COM) port on your computer. The connector needed at the computer-end of the modem cable has to fit the connector on your computer's serial (COM) port. Therefore, you will need to know the following:

1. Is your computer's serial (COM) port DSUB-9 or DSUB-25? (most contemporary computer serial ports are DSUB-9 Male). The computer-end of your RS232 cable must have the same number of terminals, as does your serial (COM) port.

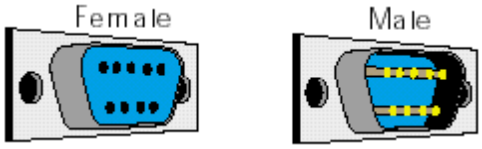


DSUB-9 (9 pins)



DSUB-25 (25 pins)

2. Note that your computer's serial (COM) port is male.



The computer-end of your RS232 cable must be female, to connect to the computer's male serial (COM) port.

+ **Hint:** Since, the KPC-3 Plus' "Computer" port has a female DSUB-25 connector, the KPC-3 Plus end of your modem cable must have a male DSUB-25 connector.

Be sure the cable between your KPC-3 Plus and the computer is correctly wired. This topic is covered when we get to the actual installation of the cable.

Technical note: *To use your KPC-3 Plus with a non-PC compatible, use a serial communication program and a serial modem cable (i.e., RS232 cable). You can get help from your computer dealer, users group and members of the amateur radio community.* ***When talking with computer dealers or others who may not be familiar with packet radio, refer to your KPC-3 Plus as an "external modem," not as a "TNC," a less familiar term.***

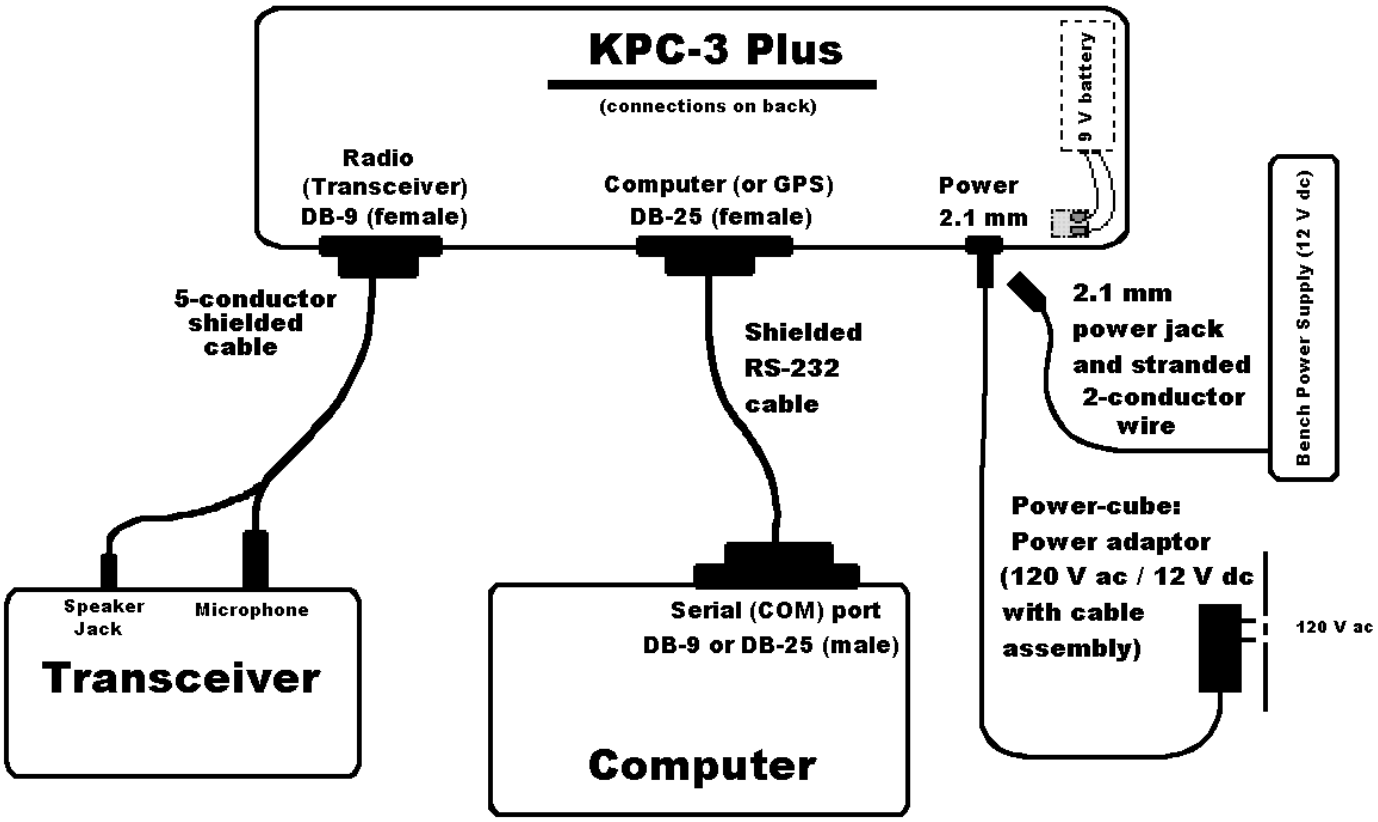
GPS Device (Optional)

The KPC-3 Plus works with Global Positioning System (GPS) devices that (1) support the NMEA 0183 standard and (2) have a dataport through which they communicate. The GPS device connects to the KPC-3 Plus through its serial ("computer") port, or alternately, through a pin on its "Radio" port.

Note: The GPS data port must be RS232 compatible to mate with the KPC-3 Plus.

How the Parts of Your Station are Connected

The following diagram shows how the components of your KPC-3 Plus packet radio station will be connected and the kinds of connectors that will be used. As you assemble your station, you may wish to refer to this diagram.



With this overview, we are now ready for step-by-step instructions for installing your station, which we will do in the following order:

- The three different ways you can power the KPC-3 Plus
- The RS232 connection between your KPC-3 Plus and your computer
- The cable assembly connecting your KPC-3 Plus to both the speaker jack and the microphone of your transceiver
- (Optional) The connection between the computer port on your KPC-3 Plus and a GPS device

As we go through the steps of setting up your station, we will shift from hardware to software issues and back, since your station needs to be set up both in terms of hardware and software.

Connect Your KPC-3 Plus to a Power Source

As illustrated on the right side of the station cabling diagram on page 33, there are three ways to supply power to your KPC-3 Plus.

- You may use external power by connecting the KPC-3 Plus to a 12 V dc source on your bench
- You may use external power by connecting the KPC-3 Plus to a 120-V ac power source using a 12-V dc adapter. Or use a 230-V ac 50-Hz to 12-V dc adapter if your mains are such.
- You may use internal power by installing a 9-V battery inside the KPC-3 Plus (as explained below, before doing this you need to install a battery clip (pig tail) inside your KPC-3 Plus).

┼ See Appendix E for the allowed range of voltage inputs.

The steps needed for each way to supply power to the KPC-3 Plus are given below, along with detailed specification for each part needed. Of course, the parts you need depend on which option(s) you use.

● The KPC-3 Plus ships with LEDs ON, so the Power indicator (LED) will turn ON when power is ON. If the LED value has been switched to OFF, the Power LED will just blink once, briefly, when power is applied.

External Power from Your Bench (12 V dc)

Here, you will use a 2.1-mm power plug and 18-22 gauge stranded 2-conductor cable to connect to a 12-V dc power supply on your bench. The steps are:

Step 2. Build a cable by attaching the supplied 2.1-mm power plug to a user-supplied 18-22 gauge stranded 2-conductor cable—connecting the center of the jack to the positive terminal of the power supply and the shell of the power jack to the negative terminal of the power supply,

Step 3. (1) Turn OFF all power, (2) attach one end of the bench-power cable to a 12-V dc power source on your bench, and (3) plug the 2.1-mm power plug into the “Power” connector of the KPC-3 Plus,

Step 4. Turn ON the power supply, (2) press the power switch on the front panel of the KPC-3 Plus to turn your KPC-3 Plus ON, (3) confirm that the power indicator next to the power switch goes on, and then (4) press the KPC-3 Plus’ power switch again to turn the power OFF.

- If the power indicator (LED) does not go on as expected, check to be sure the wiring is correct. If you have checked the wiring carefully and the power indicator still does not go on as expected, check or replace any fuses in line between the power supply and the KPC.

Caution: Do not exceed the power specifications for the KPC-3 Plus (see specifications).

If you elect to install a fuse in the positive lead, do not use a fuse of greater than 200-250 mA.

External Power Transformed from 120-V ac Mains

You can use external power from a 120-V ac power source after transforming the external power to 12 V dc (using a class 2 transformer: input 120 V ac 60 Hz 6 W; nominal output 12 V dc 300 mA).

The steps are:

Step 1. You may order from Kantronics, *as an option,* a Kantronics 120-V ac to 12-V dc 300-mA power adapter, or you may purchase the needed 120-V ac to 12-V dc 300-mA power adapter from a third party,

Step 2. Connect the 2.1-mm connector on the assembly to the power connector of the KPC-3 Plus and plug the power adapter into the 120-V ac power source (being sure that the center is positive),

Step 3. Press the power switch on the front panel of the KPC-3 Plus to turn your KPC-3 Plus ON, confirm that the power indicator next to the power switch goes on, and then press the power switch again to turn the power OFF.

+ If the power indicator (LED) does not go on as expected, the unit may be defective, so you should check with your dealer.

Internal Power, from a Battery.

For activities such as field day and back-packing, you can also install an internal 9-V battery for power, after installing a battery clip on the printed circuit board of the KPC-3 Plus.

Note: Before installing a 9 V battery inside the KPC-3 Plus, you need to install a battery clip (pigtail) and configure jumpers J1 and J2. First, purchase a 9-V battery clip (e.g., Radio Shack's heavy-duty 9-V snap connector, part number 270-324). Second, solder the clip leads to the battery pads on the printed circuit board (PCB). The pads are labeled "+BATT-" and are located in the corner of the PCB near the power connector. Make sure you solder the positive lead of the battery clip to the "+" pad, and be careful not to damage any other connections on the PCB (use a low watt iron). Third, set jumpers J1 and J2; defaults are J1 OFF (jumper on one leg only) and J2 ON (jumper on both legs). In this position, the battery will power the TNC and will be cut off (by jack P3) if an external power supply is attached. If you desire the battery to take over, i.e. run the unit, in case the external power supply is plugged in but fails (or not turned on), then set J1 ON and J2 OFF. In this configuration, whichever supply has the higher voltage (battery or external supply) will power the TNC.

To use a 9-V battery for power, install the battery as follows:

Step 1. Open the KPC-3 Plus case

Step 2. Confirm that a battery connector (pig tail, snap connector) has been installed or install one, as described above, if needed. If there is a protective clip over the battery connector, remove it

Step 3. Attach a 9-V battery

Step 4. Close the case

Step 5. Press the power switch on the front panel of the KPC-3 Plus to turn your KPC-3 Plus ON, confirm that the power indicator next to the power switch goes on, and then press the power switch again to turn the power OFF

If the power indicator (LED) does not go on as expected, the unit may be defective, so you should check with your dealer.

This concludes the installation of power to your KPC-3 Plus. The next topic is connecting your KPC-3 Plus to your computer.

Connect your KPC-3 Plus to Your Computer

Your KPC-3 Plus and your computer communicate with each other via a serial communication cable connecting the KPC-3 Plus' "Computer" port and a serial (COM) port on your computer. This connection is shown in the diagram on page 33.

Your Serial Communication Cable

The serial cable needed to connect your KPC-3 Plus to your computer (a standard off-the-shelf, RS232 modem cable) is <u>not</u> supplied with the KPC-3 Plus. You will need to purchase one, or construct one.

Note: You may construct your own cable, using wiring instructions given below.

The two options, purchasing your cable or making it, are covered next.

Purchase Your Serial Cable

To purchase your cable (from your local amateur radio dealer or computer dealer):

- specify that you need a standard serial modem cable (RS232), with high quality shielding (see Note 2 in the Radio Frequency Interference Statement section), less than 3-m in length, and at least 9 wires connected,

- specify that one connector be a male DSUB-25 connector (to connect to your KPC-3 Plus' female "Computer" port) and the other connector be a female DSUB-9 or DSUB-25 connector (depending upon whether your computer's serial (COM) port has a male DSUB-9 or DSUB-25 connector)

Make Your Serial Cable

This section is for those making your own cable instead of purchasing one.

We assume that if you are making your own cable you are familiar with the process and just need to know: (1) what parts are needed, and (2) how the parts are connected.

Parts needed:

- a male DSUB-25 connector, to connect to the KPC-3 Plus' (female) "Computer" port,
- either a female DSUB-9 or a female DSUB-25 connector, depending on the connector on your computer's serial (COM) port, to connect to the computer's (male) serial (COM) port,
- an appropriate length of high quality shielded cable with 5 or more wires,
  - as shown below, if you are using a DSUB-9 connector, either 5 or 8 wires can be connected and any other wires are unused,
  - as shown below, if you are using a DSUB-25 connector, either 5 or 9 wires can be connected and any other wires are unused.

Note: You must terminate the shield on each end of the cable assembly properly. See Note 2 in the Radio Frequency Interference Statement section.

Wiring instructions:

+ Some third-party software does not use hardware flow control and therefore does not need the RTS and CTS wires shown below.

For details on preparing your cable wiring, use the chart below that is appropriate for your configuration. In each case, there is a listing of which KPC-3 Plus ("Computer" port) pin needs to be connected to which pin on the computer's serial (COM) port. The wiring depends upon whether your computer has a 25-pin connector or a 9-pin connector, and on how many wires are in the cable (you only need 5 wires connected for KPC-3 Plus operations, but you may connect more wires to use the cable for other purposes). See page 209 for a diagram showing pin numbers on a DSUB-25 connector.

CASE 1-A: Wiring if computer has a 25-pin connector and the cable has 5 wires:

KPC-3 Plus		**Computer**	
2	<——>	2	TXD
3	<——>	3	RXD
4	<——>	4	RTS

5 <——> 5 CTS

7 <——> 7 SG

CASE 1-B: Wiring if computer has a 25-pin connector and the cable has 9 (or more) wires. Same as in CASE 1-A, and also:

KPC-3 Plus		Computer	
1	<——>	1	FG
6	<——>	6	DSR
8	<——>	8	DCD
20	<——>	20	DTR

CASE 2-A: Wiring if computer has a 9-pin connector and the cable has 5 wires:

KPC-3 Plus		Computer	
2	<——>	3	TXD
3	<——>	2	RXD
4	<——>	7	RTS
5	<——>	8	CTS
7	<——>	5	SG

CASE 2-B: Wiring if computer has a 9-pin connector and the cable has 8 or more wires. Same as in CASE 2-A, and also:

KPC-3 Plus		Computer	
6	<——>	6	DSR
8	<——>	1	DCD
20	<——>	4	DTR

Technical note for advanced users: *Two of the first five wires in each chart above (RTS and CTS) are used for hardware flow control, so if your program uses just software flow control and not hardware flow control, you may not need to use these two hardware flow control wires.*

- If you are not using a PC compatible computer, the wiring required between your computer and your KPC-3 Plus is the same wiring you would use for an external telephone modem. This cable should be available from your computer dealer. Appendix A of this manual includes details on the functions of the KPC-3 Plus' pins.

The software (terminal program) requirements are also the same as that used for a telephone modem.

Installing the RS232 Cable

Once you have purchased or made your serial communication cable, connect it as follows:

Step 1. Make sure that power to both your computer and your KPC-3 Plus is turned OFF.

Step 2. Plug the male DSUB-25 connector end of the serial cable into the “Computer” port of the KPC-3 Plus.

Step 3. Plug the female DSUB-9 or DSUB-25 connector end of the serial cable into a serial port of your computer.

Caution: Be sure you do **not** connect to a parallel port instead of a serial port: the KPC-3 Plus communicates serially, so it cannot communicate through a parallel port. Computer’s parallel ports (i.e., printer, or LPT port) usually have female DSUB-25 connectors. Do not connect the KPC-3 Plus to a female DSUB-25 connector on the computer unless you are sure it is a serial (COM) port not a parallel (LPT) port.

Step 1. Check the connectors at each end of the cable, and secure the connectors using the mounting screws,

Step 2. Turn ON the power to your computer, in preparation for installing communication software in the computer, but leave the KPC-3 Plus power OFF.

Caution: for users of a 25-pin serial modem cable: As originally shipped, the KPC-3 Plus does not connect a power source to its serial port, but an advanced user can change jumpers so that it does. If your KPC-3 Plus has been altered to apply 12 V dc to pin 13 of the computer (serial) port, this can DAMAGE your computer if pin 13 is wired. To check to see if your KPC-3 Plus has been changed in this way, see the section on jumpers in this manual. Note that the factory default setting of the jumper does not connect pin 13.

This concludes the physical installation of the serial cable needed for your KPC-3 Plus and your computer to communicate with each other.

To verify that your serial cable is correctly wired and connected, you need to install communication software in the computer and then use that software to establish communication with the KPC-3 Plus.

Configure Your KPC-3 Plus

After connecting your KPC-3 Plus and your computer with an RS232 serial modem cable, you are ready to:

- install and start terminal software in your computer to enable communication between your KPC-3 Plus and your computer,

- use the terminal communication software to establish communication with your KPC-3 Plus, including setting the speed of communications back and forth, and

- configure your KPC-3 Plus to prepare it for your use, including giving the KPC-3 Plus your CALLSIGN and setting KPC-3 Plus parameter values.

HyperTerminal

To communicate with the KPC-3 Plus, all you need is a basic terminal communication program. Windows systems (from Win95, up to WinXP) come with one called HyperTerminal. There are several other terminal programs, which can be used on Windows computers and on other operating systems. Following the instructions for HyperTerminal will give you the basics for setting up other terminal programs.

To start HyperTerminal in Windows, click Start, Programs, Accessories, Communications. HyperTerminal should be one of the possible selections. If it is not, it may not have been installed when Windows was installed. You will then have to use the Add Program, Add Windows Component from the Control Panel to install it.

Click HyperTerminal from the Accessories menu. HyperTerminal will open with a window asking for a name for a new connection. Enter “KPC-3Plus9600”. Pick a different icon if you want, then click OK.

A “Connect To” window will now open. Use the drop down menu under “Connect using:” to select COM1. Then click OK.

A “COM1 Properties” window will now appear. Select 9600 bits/s, 8 data bits, no parity, 1 stop bit, and hardware flow control. Click OK.

You should now be at the main HyperTerminal screen. Click “File”, then “Properties”. In the Properties window, select Settings. “Terminal keys” and “CTRL+H” should already be selected. Use the drop down menu under “Emulation:” to select “TTY”. “Terminal Setup” will allow you to change your cursor; “Use destructive backspace” should be checked. Under “ASCII Setup” the defaults are sufficient (delays of 0 and only “wrap lines” checked). Click OK to close the Property windows.

You should now be back at the main HyperTerminal screen. Click File, then Save. The next time you click Start, Programs, Accessories, Communications, you will see that there is a HyperTerminal menu with a selection for “KPC-3 Plus 9600.ht”. Clicking this will get you immediately to the main HyperTerminal screen already set up for the KPC-3 Plus.

Later you may wish to change the baud or com port you are using with the KPC-3 Plus. Before you can do that in HyperTerminal, you will first have to tell HyperTerminal to “disconnect”; there is a button on the HyperTerminal screen to do that, looking like a phone being hung up. Once disconnected, you can use the Properties menu to change the HyperTerminal configuration.

Now turn on the KPC-3 Plus.

AUTOBAUD

The first time your KPC-3 Plus is used, it runs an AUTOBAUD routine to assure that the KPC-3 Plus baud is the same as that of whatever communication software you are using.

The KPC-3 Plus' AUTOBAUD routine works as follows:

Step 1. AUTOBAUD sends and resends the message "PRESS (*) TO SET BAUD" at one baud after another. When the KPC-3 Plus' baud matches that of your terminal program, you can read this on your screen; if the two rates do not match, the message appears garbled, if it appears at all.

Step 2. Any time while this message is being sent, whether it is readable or garbled on your screen, enter the "*" character (SHIFT+8 key combination).

Step 3. When the AUTOBAUD routine receives, the "*" character from the serial port, it uses that character to determine the baud in use.

Step 4. AUTOBAUD then sets the KPC-3 Plus' baud (ABAUD) to match the baud used.

Technical note: *The KPC-3 Plus' Baud is stored in the KPC-3 Plus as the value of the ABAUD parameter, so when you use the KPC-3 Plus later, it will start with that baud. This is made possible by an internal lithium battery supplied with the KPC-3 Plus. Later, you can change the TNC baud if you want, as long as you change your terminal's baud to match.*

Step 5. Finally, AUTOBAUD sends a "sign-on" message and asks for the user's CALLSIGN, which will also be stored in the KPC-3 Plus and used until changed. At this point, the user is ready to give commands to the KPC-3 Plus.

+ **Hint:** Do not confuse the KPC-3 Plus' baud for serial communication on the ***computer*** port with the KPC-3 Plus' 1200-baud for communicating using the ***radio*** port, which is connected by a custom-wired cabling to a transceiver.

Setting Basic Communication Parameters

There are several KPC-3 Plus settings in addition to ABAUD (baud) and MYCALL (CALLSIGN) that you may wish to change, especially if parameter values in your KPC-3 Plus have been changed from their default settings. Even if you do not want to change anything now, knowing about these parameters and knowing how to reconfigure your KPC-3 Plus gives you more control in using it with HyperTerminal.

Required TNC Parameter Settings

The following TNC parameter settings are needed:

- COMMAND = $03 (CTRL+C) (default = $03, so unless this has been changed, you do not need to do anything).
- CANLINE default = CTRL+X (HEX $18)
- CANPAC default = CTRL+Y (HEX $19)
- PASS default = CTRL+V (HEX $16)

Optional Parameter Settings

The following TNC parameters are generally used with their default settings when running HyperTerminal, but you may wish to change them. If they have been set to non-default values, you may wish to change them back.

- ECHO = ON (default)
- FLOW = ON (default)

  Technical note: *Usually, ECHO and FLOW go together, both ON or both OFF. When using software that supports split screen displays, for example, set both ECHO and FLOW to OFF.*

- XFLOW = ON (default)

  Technical note: *XFLOW controls whether or not software flow control is ON. If XFLOW is set to OFF, the following parameters should also be set to 0 (zero): XON, XOFF, START, and STOP.*

- FILTER = OFF (default)
- 8BITCONV = ON (default)

┼ **Hint:** Also, the **MONITOR** parameter may be ON or OFF (default = ON). You may have Monitor ON while using HyperTerminal. But to prevent a build-up of data in an internal buffer in the TNC, you should turn Monitor OFF when exiting HyperTerminal **and** leave the TNC ON. This is because when the TNC's receive buffer is full, your station will give a **BUSY** signal to other stations that try to connect.

Trouble-Shooting Difficulties in Communicating

If you are having difficulty establishing and maintaining communication between HyperTerminal and your TNC, here are some points to consider.

Problem: Go to Terminal Screen, but Nothing Happens

One possibility is that the current COM port does have a device on it, but it is not a TNC. It might be a mouse or an internal phone modem, for example. To see if it is a mouse, go to the TERMINAL Screen, start moving your mouse and see if sequences of characters appear as you move the mouse. If so, change the COM port setting in HyperTerminal and try again.

Another possibility is that the KPC-3 Plus has already been programmed with a different baud than the one set in the terminal program. You can try changing the baud in the terminal program, or you can use the self-test jumper (J11) to erase the current settings in the KPC-3 Plus, forcing the KPC-3 Plus to try its Autobaud routine.

Problem: You Are Getting Bad or Intermittent Data

Many PC compatible computers are designed with a built-in potential conflict that may cause a problem. Both COM2 and COM4 (and both COM1 and COM3) may use the same interrupt (IRQ) mechanism to get the attention of the computer. This means that if devices on both COM2 and COM4 (or both COM1 and COM3) are sending signals to the computer at the same time, this can cause problems for programs, such as HyperTerminal, that use either COM port. If you are getting bad data, check to see if you have another device that might be getting confused with your TNC (on COM4 if your TNC is on COM2; COM3 if your TNC is on COM1) and make it inactive or remove it while using HyperTerminal with your TNC.

Problem: Your TNC Stops Behaving Normally

HyperTerminal is intended for use with a TNC in NEWUSER or TERMINAL Interface Mode (or, less often, in BBS or GPS Interface Mode), but it may receive data from the TNC in any Interface Mode. Non-response, garbled data, or partially garbled data may indicate the TNC has been switched to HOST or KISS mode and needs to be switched back to NEWUSER or TERMINAL. See the TNC documentation for instructions on how to determine what Interface Mode your TNC is in currently and how to change it to the Interface Mode you want (NEWUSER or TERMINAL).

Connect the KPC-3 Plus to a Transceiver

The last connections you need to establish are between your KPC-3 Plus and your transceiver. You will construct and install a transceiver cable assembly that has two functions: (1) to send signals from the speaker jack of your transceiver to your KPC-3 Plus' "Radio" port for processing, and (2) to send signals from your KPC-3 Plus' "Radio" port to the microphone connection of your transceiver for transmission.

As illustrated in the diagram on page 33, the transceiver cable connects to the "Radio" port on the KPC-3 Plus via a male DSUB-9 connector shipped with your KPC-3.

This section goes step-by-step, dealing with the following questions:

- What parts will be needed for the assembly?

- What are the exact wiring requirements for connecting your transceiver to the KPC-3 Plus, as determined by information given in this document ***and also*** by requirements for your own particular make and model of transceiver (as determined by the documentation for your transceiver)?

- What are the steps used in constructing and connecting the cable assembly?

- Once the cable is in place, how does one set the volume level of the transceiver for optimal performance?

Parts for Connecting the KPC-3 Plus to a Transceiver

The following parts (shipped with your KPC-3 Plus unless otherwise noted) are needed for constructing the cable assembly that will connect your KPC-3 Plus and your transceiver:

- DSUB-9 connector kit (including hardware), to make the DSUB-9 connector for the end of the transceiver cabling that connects to the KPC-3 Plus' "Radio" port

- A 3-foot (0.91 m) length of 5-conductor shielded cable, to connect a user-supplied microphone connector (i.e., the Mic plug) and/or a user-supplied auxiliary connector (plug) or external audio connector (plug) for your transceiver to the DSUB-9 connector supplied with your KPC-3 Plus

Preparing the Transceiver Cable Assembly

This section gives instructions for how to assemble the cable assembly needed to connect your transceiver to your KPC-3 Plus, starting with the wiring requirements.

Wiring Directions (Connecting KPC-3 Plus and Transceiver)

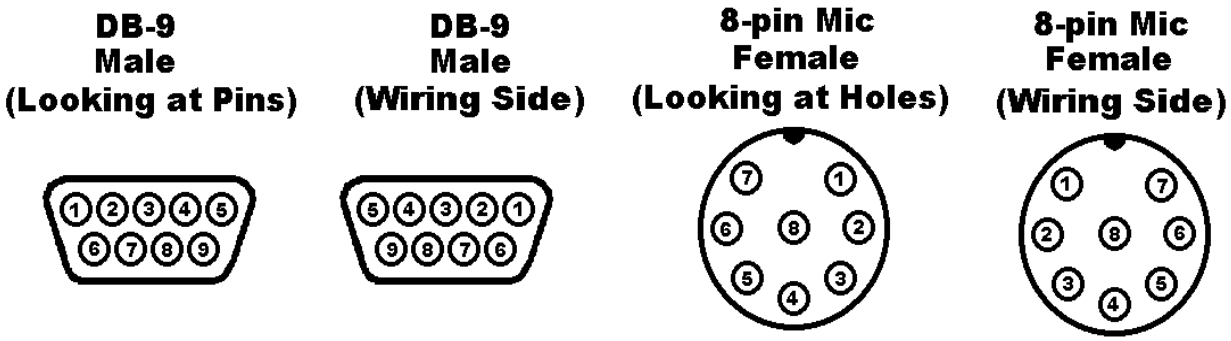
You will need to determine which pins on your DSUB-9 connector will be wired to which connections on your particular transceiver's microphone connector and speaker jack. Wiring directions are covered after the following examples of typical transceivers, including handhelds.

♦ There are many different models of transceivers, each with their own exact requirements for how they are to be connected to devices such as a KPC-3 Plus. While the following examples will help, you will need to refer to your transceiver documentation (or transceiver dealer) for exact instructions on which connections you need to make as you wire your transceiver cable assembly.

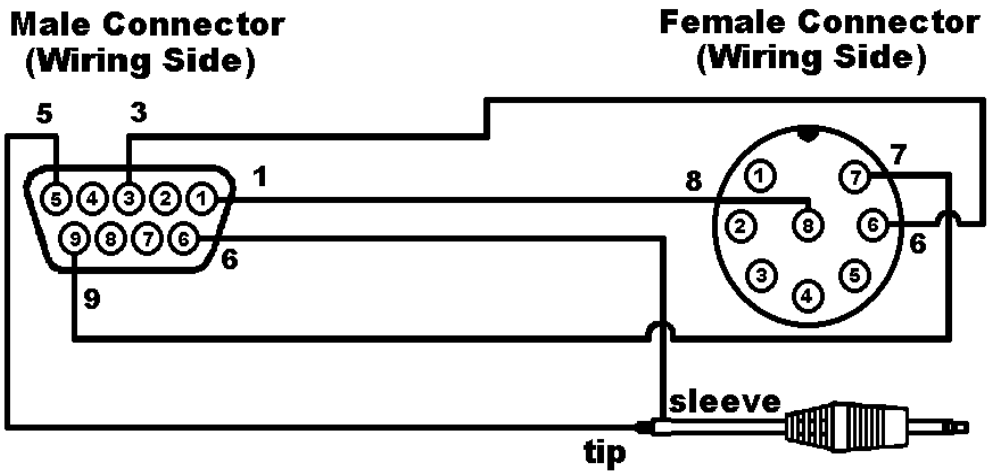
Note: If you need further assistance on your radio cabling, contact the radio manufacturer, your dealer, or a local amateur who has already installed a TNC. Kantronics is not likely to have pin assignments for specific transceiver models.

The following diagrams, used for example only, show wiring connections between the male connector for the KPC-3 Plus' "Radio" port, and common transceivers (including HTs) from three major manufacturers: Yaesu, Icom, and Kenwood.

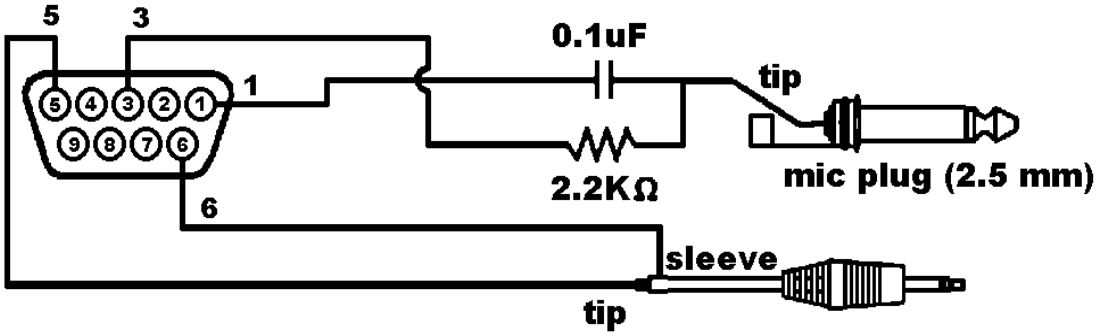
+ These diagrams may also apply to transceivers from other manufacturers, but you need to check to be sure how your transceiver needs to be wired. See the section on “Interfacing Hand-Held Radios” in the “Advanced Installation” appendix for further information.



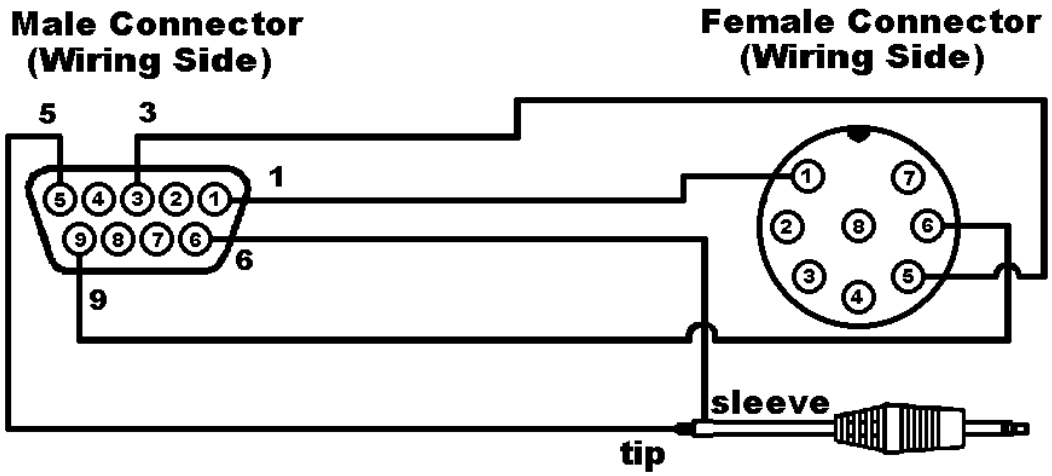
Yaesu Wiring



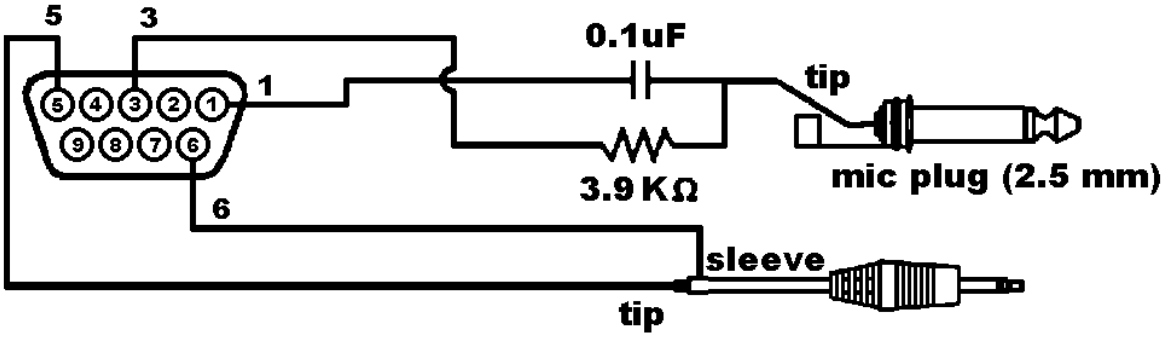
Yaesu Hand-Held Radios



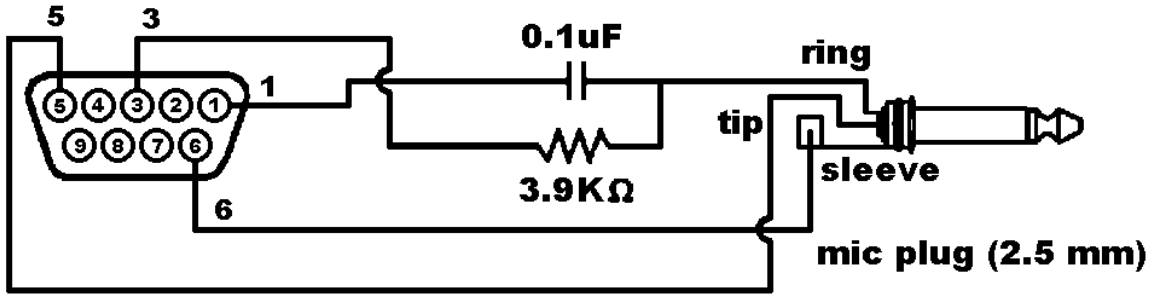
Icom Wiring



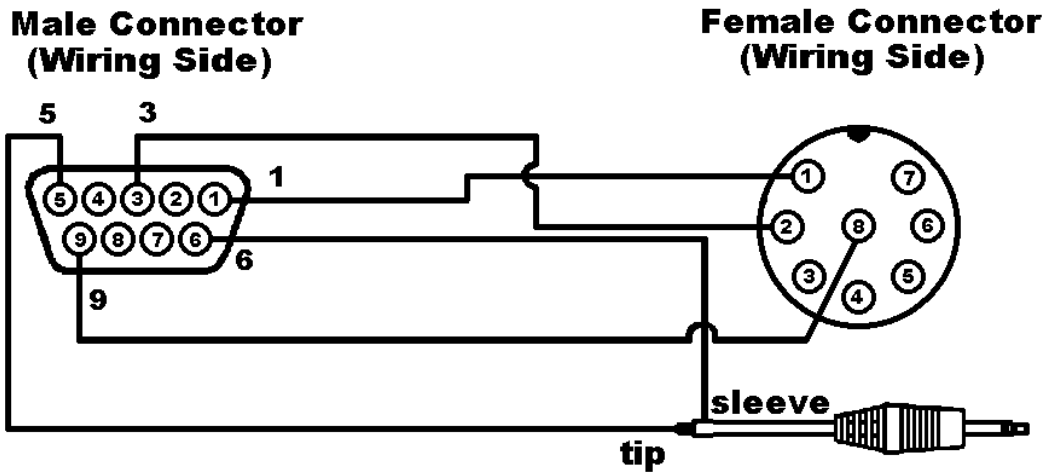
2 AT Style Hand-Held Radios



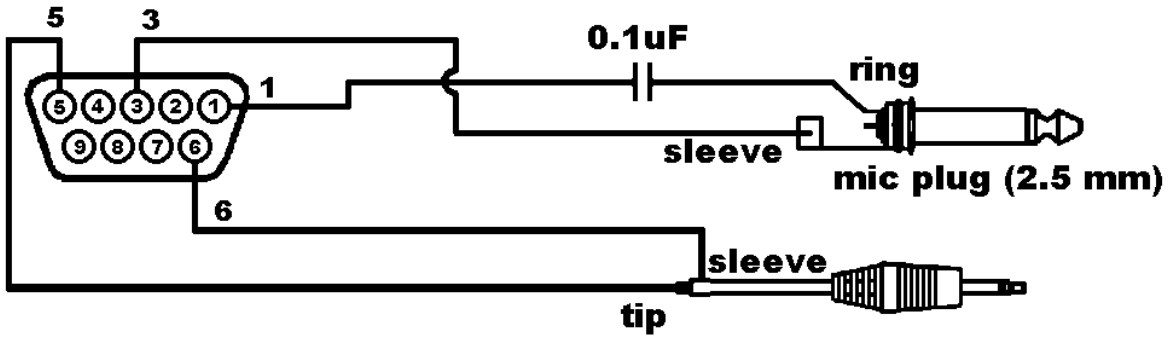
W2A Style Hand-Held Radios



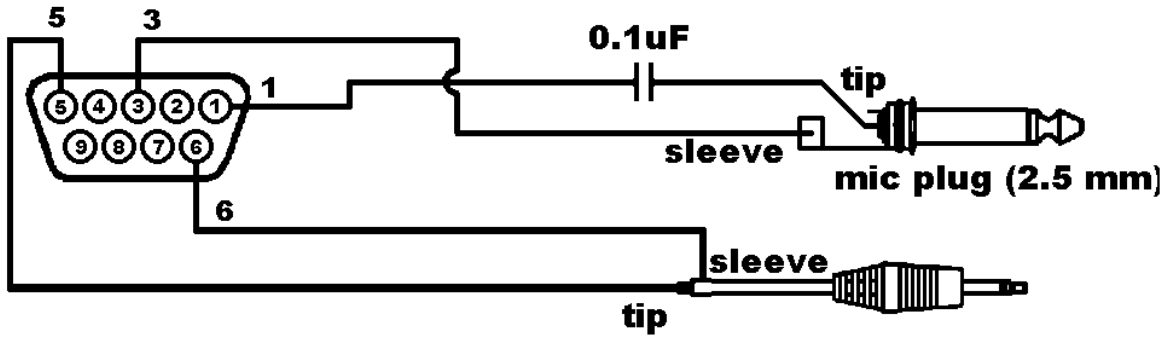
Kenwood Wiring



TR-2600 and later Hand-Held Radios



TR-2500 Hand-Held Radios



Your Transceiver's Microphone Connections to the KPC-3 Plus.

For the microphone on your transceiver, you may wish to make notes of the following connections, which you will use, for wiring your assembly:

- identify and make a note of the microphone input (which will be wired to pin 1, **transmit audio** on the KPC-3 Plus' DSUB-9 "Radio" port),

- identify and make a note of the PTT or STBY connection (which will be wired to pin 3, **Push-to-Talk** on the KPC-3 Plus' DSUB-9 "Radio" port),

- for base or mobile (but not hand-held) transceivers, identify and make a note of the ground connection (which, optionally, may be wired to pin 9, one of the **ground** pins on the KPC-3 Plus' DSUB-9 "Radio" port, or left unconnected).

Your Transceiver's Speaker Jack Connections to the KPC-3 Plus

For the speaker jack on your transceiver, make a note of the following connections, which you will use for wiring your assembly:

- identify and make a note of the connector to your external speaker plug (which will be wired to pin 5, **receive audio** on the KPC-3 Plus' DSUB-9 "Radio" port),

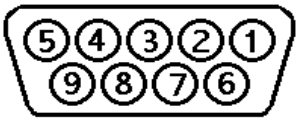
- identify and make a note of the connector to your external speaker plug (which will be wired to pin 6, one of the **ground** pins on the KPC-3 Plus' DSUB-9 "Radio" port).

Constructing the Cable Assembly

As you construct the cable assembly, it may be helpful to refer to the diagram on page 33.

Important: As shown in the diagram immediately below, looking at the rear panel, the pins in the KPC-3 Plus' female DSUB-9 "Radio" connector are numbered from the ***upper right*** (pin 1) to the ***lower left*** (pin 9). As you connect wires to pins on the DSUB-9 connector on your cable, you need to be sure to connect to the correct pins.

As shown in the diagram below, if you look at the solder side (wiring side) of the male DSUB-9 connector, the pins are numbered (again) from ***upper right*** (pin1), to ***lower left*** (pin 9).



+ Similarly, when working with a typical 8-pin Mic connector where you know the pin numbers of pins, you need to consider which way you are looking at the connector (looking at pins/sockets or looking at the wiring side, where the wires are actually connected).

+ **Hint:** Make notes of which wires (colors, cables) you use for each connection, so you will be able to keep track of which connection is which.

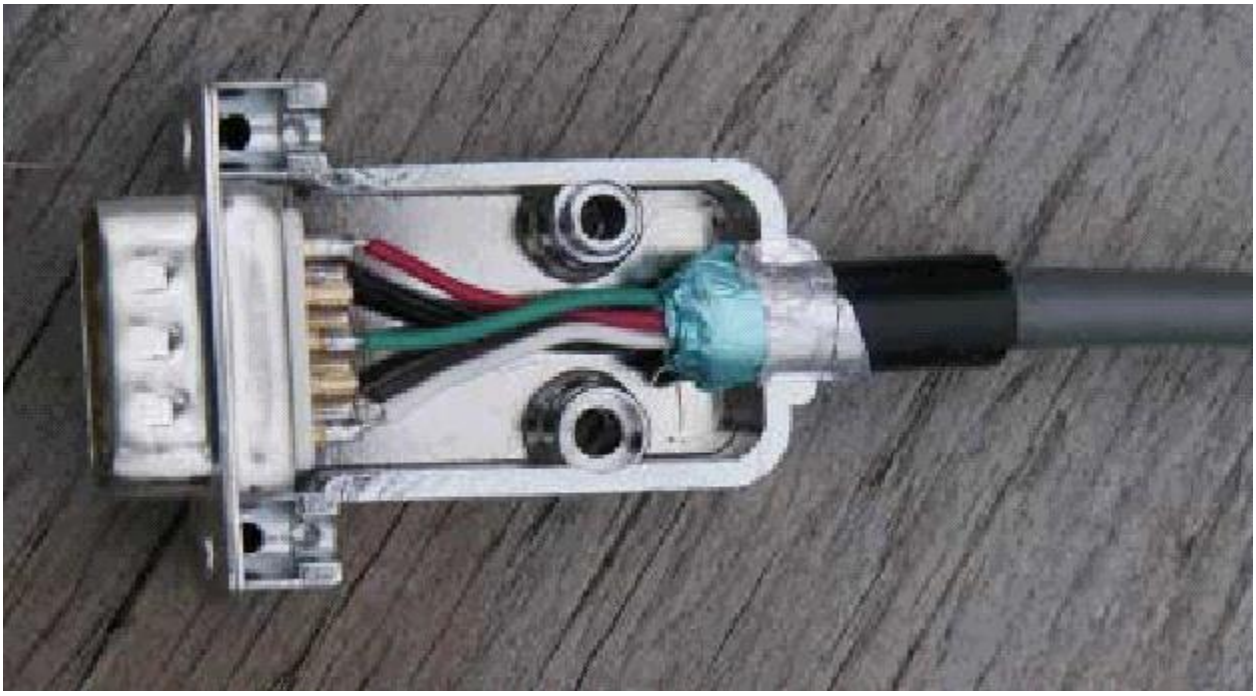
To construct your transceiver cable assembly, proceed as follows:

+ **Hint:** You may find it easier to wire the male DSUB-9 connector if you first connect it to the KPC-3 Plus and use the KPC-3 Plus as a “jig.” This may also keep the male pins straight if you apply too much heat and soften the plastic in the male connector.

A shielded cable must be used with the shield properly terminated 360° to the connector.

Instructions for making a cable assembly using the supplied foil shielded cable and metalized plastic back shell follow:

1. Strip cable outer covering back enough to expose enough metalized plastic foil. Do not cut or rip foil.
2. Carefully loosen the drain wire and foil from around the insulated wires.
3. You will need to build up the diameter of the cable, where it enters the back shell, with any type of tape or sleeve, so that when the metalized plastic shell halves are screwed together they will hold the cable as strain relief and press against the metal foil.
4. Discard the metal strain relief pieces and screws that go with them—they are of no use.
5. Connect wires as needed to terminals of connector. Keep track of which wire is connected to which terminal.
6. Carefully fold the metalized foil back over the cable and Z-fold it so the metal side of the foil is exposed to the connector back shell at the entrance area. Spread the foil over the cable so you get as much coverage as possible (you will not get 360° coverage, but do the best you can).
7. Place the drain wire in the gap where you can not get foil coverage.
8. Place one of the shaped washers on each of the long, half-threaded screws, so they are next to the head of the screw (with the bent ends of the washers facing away from the heads of the screws).
9. Now place the metalized plastic shell halves together over the connector and cable, with the (2) two long half-threaded screws and two shaped washers, to attach the whole assembly to the KPC-3 Plus.
10. When you screw the two halves together there should be enough compression to hold the cable in place and there should be solid contact between the metal side of the foil and the metalized back shell. See figure below.
11. If your radio is equipped with a single metal connector, such as the Kantronics dvr 2-2, then this same cable with connector attachment technique should be used. Otherwise follow the wiring directions that follow.



Wiring between your Transceiver Microphone and the KPC-3 Plus:

Follow the notes taken above and wire the cable assembly in the following order:

Step 1. Using a user-supplied microphone connector appropriate for your transceiver and the notes taken above, attach wires as follows:

- Attach the wire from pin 1 (transmit audio) on the DSUB-9 to the microphone connector's input pin,
- Attach the wire from pin 3 (push-to-talk) on the DSUB-9 to the microphone connector's PTT or STBY pin,
- (Optional) for base or mobile units, attach the wire from pin 9 (one path to ground) on the DSUB-9 to the ground pin on the microphone connector.

Connecting your Transceiver's Speaker Plug to the KPC-3 Plus

Use an appropriate third party speaker plug and the notes taken earlier to do the following wiring to the DSUB-9 cable assembly:

Step 1. Locate the wire that connects to the audio of the external speaker plug and attach it to the wire from pin 5 on the DSUB-9 (Receive Audio)

Step 2. Locate the wire that connects to the ground of the external speaker jack and attach it to the wire from pin 6 on the DSUB-9 (Ground)

This is all you need to do to wire the speaker plug part of the assembly.

Important: Keep your wiring notes for future reference.

This completes the construction of your transceiver cable assembly. Now it needs to be connected.

Connecting the Transceiver Cable Assembly

Now you are ready to use this cable assembly to finish assembling your packet radio station.

Step 1. Check to be sure all the parts of your completed cable assembly are properly fastened and ready for connection to your transceiver and your KPC-3 Plus.

Step 2. Turn OFF all power.

Step 3. Plug the male DSUB-9 connector on the cable into the female DSUB-9 connector on the KPC-3 Plus' "Radio" port,

Step 4. Secure the male DSUB-9 connector on the cable to the female DSUB-9 connector on the KPC-3 Plus' "Radio" port, by screwing the two long, half-threaded screws into the threaded nuts on each side of the KPC-3 Plus' "Radio" port,

Step 5. Plug both the microphone connector and the external speaker jack into your transceiver.

Adjusting the Receive Volume of Your Transceiver

To adjust the receive volume of your transceiver to the proper level (or to confirm that it is already set at the proper level), do the following:

Step 1. Turn ON your computer (with your transceiver OFF),

Step 2. Start your terminal program,

Step 3. Turn ON your KPC-3 Plus,

Step 4. If not already done, set the KPC-3 Plus' baud (ABAUD) and CALLSIGN,

Step 5. Turn ON your transceiver and open the squelch control (turn it fully counter-clockwise),

Step 6. Slowly turn the transceiver's receive volume up until the RCV LED (receive indicator) on the KPC-3 Plus lights up (turns ON),

Step 7. Increase the transceiver's receive volume control slightly above the point at which the KPC-3 Plus' RCV LED lit up (turned ON),

Step 8. Slowly turn the transceiver's squelch control clockwise until the KPC-3 Plus' RCV LED goes OUT (turns OFF),

Now the receive volume control on your transceiver is properly adjusted for sending signals to your KPC-3 Plus.

Your KPC-3 Plus is now installed and ready to use. You may now go to the “Getting Started” chapter, to learn how to do the basic operations with your KPC-3 Plus.

- If you do not see the receive LED light, check the cabling between the radio and the KPC-3 Plus. Also, until it has been initialized from the computer, the KPC-3 Plus will not show ANY indication of receiving.

- If you connected to a fixed level (unsquelched) receive audio from the radio, the receive LED may remain on all of the time. In this case, set the parameter CD to SOFTWARE. Then, the receive LED will not light until valid packets are heard. (see CD in the commands section)

Transmit level adjustment

The 1200-baud data drive level (transmit audio voltage, TXA) from the KPC-3 Plus to your radio is adjusted digitally, using your keyboard and the transmit level (XMITLVL) or the calibrate (CAL) command. Previous models used a set of jumpers and an analog potentiometer (pot) to set this level. With the KPC-3 Plus, no jumpers are used, just a command.

You should adjust the drive level to provide about 3 to 3.5 kHz of deviation with your radio. If you don’t have access to a deviation meter, adjust drive so your packet transmissions generate about the same audio sound as packets heard from other stations.

The XMITLVL command has a parameter default value of 100; this corresponds to a drive voltage of about 50 mV. You may change this level by increasing the parameter/count. For counts below 256, the voltage is increased in 0.5 mV steps; above that, the drive increases roughly 15 mV per step.

- For details, see the description of the XMITLVL command in the “Command Reference.”

Alternatively, you may use the calibrate command (CAL) which has a zoom feature; that is, you can hold down the ‘-’ or ‘+’ key to decrease or increase the voltage (as you watch a meter or listen). See the commands section for more detail on these commands. CAL is not available in the NEWUSER mode; you must be in TERMINAL mode to access CAL.

The next section of this chapter is for those who wish to connect a GPS device to their KPC-3 Plus. Skip this section if it does not apply to you.

Connecting to a GPS Device (Optional)

To use your KPC-3 Plus with a GPS device, you can connect the data output from the GPS device to either the DSUB-25 “Computer" port (i.e., serial port) on the back of the KPC-3 Plus or alternately, to a pin on its radio port.

Since you need to remove the computer’s serial modem cable from the KPC-3 Plus’ “Computer” port and replace it with the GPS’s cable before you can use the GPS device with the KPC-3 Plus, you will need to use your computer to configure your KPC-3 Plus to work with the GPS device before you connect your GPS device to your KPC-3 Plus.

Note: You can attach both the KPC-3 Plus and a GPS device to your computer at the same time, using an APRS HSP (hardware single port) cable and the APRS program.

Since cabling may vary from one GPS vendor to another, the cable is not supplied with your KPC-3 Plus. You will need to purchase or make a cable, using the specifications in your GPS manual.

To connect the GPS receiver and the KPC-3 Plus as a “Stand-alone” position reporting system without a computer, you will need to connect a minimum of 2 wires for data flow from the GPS receiver to the KPC-3 Plus. Connect the GPS receiver “DATA OUTPUT” to TXD (pin 2) on the KPC-3 Plus. Connect the GPS receiver “SIGNAL GROUND” to signal ground (pin 7) on the KPC-3 Plus. Your GPS receiver may require other pins to be connected, check its documentation.

An alternate input is available in the KPC-3 Plus (at firmware version 8.3 or later) for GPS data, on its “Radio” port. The alternate GPS data input is enabled with the command GPSPORT, and uses pin 2 on the KPC “Radio” port.

See your GPS manual and the chapter in this manual on “GPS NMEA Interfacing Capability” for details on using your GPS device with your KPC-3 Plus based packet radio station.

┼ Note: some weather station devices have data outputs similar to GPS devices. The configuration of the KPC, and connection to the device is the same.

This concludes the installation of your station.

♦ For more information on installation topics, see the appendices.

Getting Started

This chapter gives you important information and examples to get you started using your packet radio station.

We assume you have installed your station and familiarized yourself with the use of with your KPC-3 Plus, as covered in the chapters on “Installing your KPC-3 Plus”.

If you have not yet installed your station, we suggest that you do so before continuing with this chapter.

Material covered here includes diagrams, explanations, and mini-tutorials on:

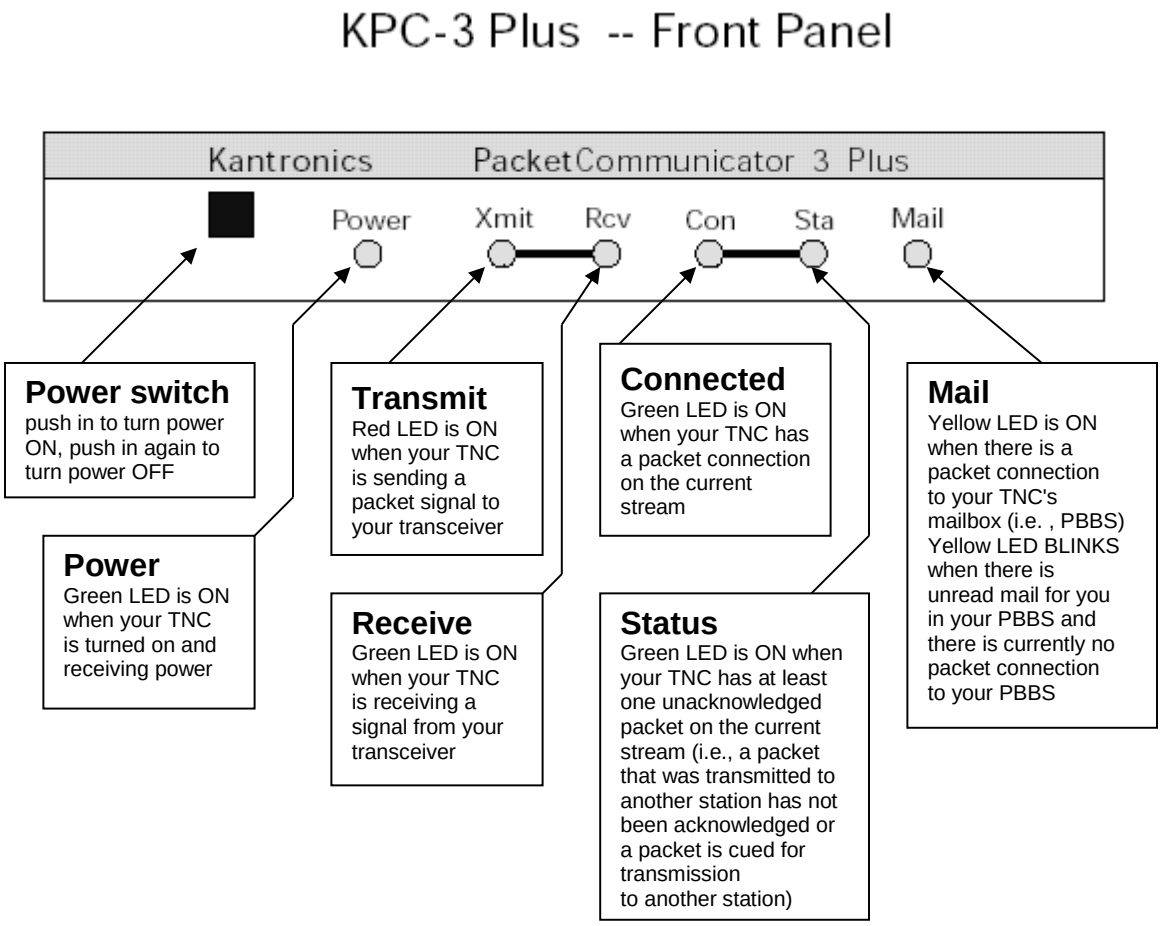
- the front panel of the KPC-3 Plus,
- beginning a session,
- giving commands and transmitting data,
- the 22 NEWUSER commands that you will be using most of the time,
- using selected NEWUSER commands,
- using the Personal Bulletin Board System that is in your KPC-3 Plus,
- monitoring transmissions from nearby packet radio stations, and
- connecting to another station and initiating a conversation.

♦ Reading this chapter (or just skimming the material, if you are already familiar with the topics) will also help you use the following chapters on different modes of use of the KPC-3 Plus, the “Command Reference” chapter, and any appendices that are relevant to your needs.

The Front Panel of the KPC-3 Plus

You have already used the POWER switch on the KPC-3 Plus to turn power on and off. This is a good time to get familiar with all the indicators on the front panel of the KPC-3 Plus.

The following diagram shows the placement and function of the controls and indicators on the front of your KPC-3 Plus, as well as a brief explanation of each:



Beginning a Packet Session

The steps involved in starting a session are covered in detail in the “Installing your KPC-3 Plus” chapter.

Start HyperTerminal, select the KPC-3 Plus icon, and turn on the KPC-3 Plus.

Once your initial setup is completed (including giving a CALLSIGN to your KPC-3 Plus and setting other parameters as needed), the first thing you will see each time you start

a new session is a sign-on message from the KPC-3 Plus (sent to HyperTerminal and displayed on your computer screen) such as the following:

```
KANTRONICS KPC3PMX VERSION 9.1
(C) COPYRIGHT 2002 -2005 BY KANTRONICS INC. ALL RIGHTS RESERVED.
DUPLICATION PROHIBITED WITHOUT PERMISSION OF KANTRONICS.
cmd:
```

The last line of this message ("cmd:") is the KPC-3 Plus' report that it is now in "Command" mode, which means that it is now expecting information from HyperTerminal to be commands for it (the KPC-3 Plus) to interpret and carry out. Before learning about the specific commands you can give the KPC-3 Plus, it is important to keep in mind the difference between giving the KPC-3 Plus commands and giving the KPC-3 Plus data that is to be transmitted.

Giving Commands and Transmitting Data

The KPC-3 Plus will react to data received from HyperTerminal (or any other source) either as a command to follow or as information to transmit.

Command Mode

When in the "COMMAND" mode of communication, the KPC-3 Plus will interpret data received from HyperTerminal (or any other source) as being a command to process, not as data to transmit. To instruct your KPC-3 Plus to connect, to disconnect, or to change any of the operating parameters in the KPC-3 Plus you must be in the COMMAND mode.

- When placed in COMMAND mode (and ready to receive the next command) the KPC-3 sends a "cmd:" prompt for display on the computer's screen.

Converse (Conversation) Mode

When in "CONVERS" (conversation) mode, the KPC-3 will interpret data received from HyperTerminal (or any other source) as data to transmit. In "CONVERS" mode, the KPC-3 Plus interprets most characters as information to be transmitted, while interpreting certain pre-defined characters (which the user can change) as control characters rather than as data to be sent. For example, a "backspace" character causes a backspace and is not transmitted.

- The KPC-3 Plus will automatically switch to the CONVERS mode for you when you connect to someone or when someone connects to you and will return to the COMMAND mode when a disconnect occurs. When you are in COMMAND mode, you may switch to CONVERS mode just by giving the command K (or CONVERS). If you are in the CONVERS mode and want to force your KPC-3 Plus back to the COMMAND mode, you type a <Ctrl+C>.

TRANS (Transparent) Mode

A second way to transmit data is to instruct the KPC-3 Plus to ignore the “control characters” (e.g., “backspace”) and just transmit every character as data. This is called the “TRANS” (transparent) mode of communication. For example, if data received by the KPC-3 Plus for transmission in TRANS mode includes “backspace” characters (i.e., characters that would cause a backspace in CONVERS mode), the KPC-3 Plus will transmit the backspace characters as part of the data.

- Before you can switch to TRANS mode, you need to set INTFACE to TERMINAL instead of NEWUSER. (This is because, as described in the next section, the TRANS command is not a NEWUSER command.) To get out of TRANS mode and back to COMMAND mode, enter “<Ctrl+C> three times, with a pause of less than one second between each entry.

NEWUSER Commands

You have a choice of whether to have your KPC-3 Plus prepared to deal with a small set of (22) NEWUSER commands (which is the default setting) or with the full set of over 130 commands it is capable of carrying out.

- Note: To switch to the full command set of the KPC-3 Plus, get in COMMAND Mode (<Ctrl+C>) and give the command INTFACE TERMINAL. To switch back, give the command INTFACE NEWUSER.

List of NEWUSER Commands

The following is an alphabetical listing of the 22 NEWUSER commands and a brief description of each.

BKONDEL - When ON echoes a backspace-space-backspace sequence to the screen each time you press the DELETE key to erase errors from your screen. When OFF, echoes a \ for each press of the DELETE key.

CONNECT - Used to establish a packet connection with another station.

CONVERS - Places the KPC-3 Plus in CONVERS mode.

DISCONNE - Used to disconnect from a station you have been talking to.

DAYTIME - Used to read or set the clock in the KPC-3 Plus.

DELETE - Determines the character received from the computer which will cause the previous character to be removed from the transmit buffer.

DISPLAY - Causes a screen display of all KPC-3 Plus parameters.

DWAIT - Time delay imposed to insure that digipeaters have priority access to the radio frequency.

ECHO - When ON characters you type on the keyboard will be echoed back to your terminal. When OFF characters you type will not be echoed.

HELP - Displays a complete list of all available commands on your screen.

INTFACE - Selects the NEWUSER command set or the complete (TERMINAL) command set. Also used to place the KPC-3 Plus in HOST, BBS, KISS, XKISS, MODEM or GPS mode.

NEWUSER Commands

NEWUSER commands are a subset of the full TERMINAL command set. NEWUSER commands are all you need for basic, beginner operations of your packet radio station. These commands are explained in the **Command Reference** section of the manual.

Select the current interface mode
INTFACE: Set the current interface mode to NEWUSER, TERMINAL, BBS, HOST, KISS, XKISS, GPS or MODEM. Default is NEWUSER, which makes available just the basic commands listed here.

Get help and other information about your KPC-3 Plus
HELP: List all available commands or a brief description of a specified command.
DISPLAY: Show current values for all parameters or for a specified group of parameters.
VERSION: Show the version number of the EPROM (firmware) installed in your KPC-3 Plus.

Configure your personal mailbox (PBBS)
MYPBBS: Change the callsign of your PBBS.
PBBS: Change the size of your PBBS (i.e., RAM used).

Configure your KPC-3 Plus
RESET: Restart the modem -- equivalent to turning your KPC-3 Plus

Define your KPC-3 Plus' use of characters sent to it from your computer
ECHO: Display characters you type (default is ON, for use with full duplex communication programs).
BKONDEL: Select the effect of entering a "delete" (i.e., erase the previous character or display a "\").
DELETE: Change which key your KPC-3 Plus will interpret as "delete" (default is backspace key).

Display communication activity
MONITOR: Display packets from other stations (default is ON).
STATUS: Display current I/O stream and link status of other connected streams.
MHEARD: Display a list of stations recently heard by your KPC-3 Plus.

Session control
CONNECT: Start a session (i.e., connect to a station) and automatically switch to CONVERS Mode.
DISCONNE: End a session (i.e., disconnect from a station).
UNPROTO: Set a destination and digipeater path for unproto packet transmissions.

Switch from giving your KPC-3 Plus commands to using it to send data
CONVERS: Switch from COMMAND communication mode to CONVERS (i.e., conversation) Mode. The default way to go *back* to COMMAND Mode is to enter <Ctrl+C>.
K: Same as **CONVERS.**

K - Switches the KPC-3 Plus into the CONVERS mode (same as CONVERS).

MONITOR - When ON allows the KPC-3 Plus to display monitored packets on the screen. When OFF, no packets will be displayed unless someone actually connects to you.

MHEARD - Displays a short list of stations that have recently been heard by your KPC-3 Plus.

MYCALL - Used to set the callsign of your KPC-3 Plus.

MYPBBS - Sets the callsign used for your personal mailbox in the KPC-3 Plus.

PBBS - Used to set the size (kbytes) of the Personal BBS in your KPC-3 Plus.

RESET - Causes the KPC-3 Plus to perform a soft reset.

STATUS - Displays the status of the current stream and all connected streams.

TXDELAY - Sets the time delay between Push-to-Talk and the beginning of data.

UNPROTO - Sets the destination field and digipeaters used for any unconnected data packets transmitted.

VERSION - Displays the current version number of the EPROM (firmware) in your KPC-3 Plus.

Using NEWUSER Commands

A brief look at the use of selected NEWUSER commands will give you the familiarity you need to continue on your own using KPC-3 Plus commands.

Check Your KPC-3 Plus' Version Number and ID

To check your KPC-3 Plus' version number and ID all you need to do is ask to see the current setting of the command called VERSION:

Step 1. Go to COMMAND mode (if you are not already there).

Step 2. Type "VERSION" (or the short form, "V") and press the ENTER key to enter the command.

Step 3. The KPC-3 Plus will send its version number and other ID information for display on your monitor.

Get Help

To see a one-line explanation of any command that is currently available:

Step 1. Go to COMMAND mode (if you are not already there).

Step 2. Type "HELP," (or "?") followed by one or more spaces and the name of the command for which you want help.

Step 3. The KPC-3 Plus will send a short "HELP" message about the command, for display on your monitor.

View Current Values of Parameters

To see the current value of parameter(s) for any variable (other than the “immediate” commands, like CONVERS, that are simple actions, not settings):

Step 1. Go to COMMAND mode, as described above (if you are not already there).

Step 2. Type the name of the command for which you want to see current parameter value(s).

Step 3. The KPC-3 Plus will send a message consisting of the name of the command and the value(s) of its parameters for display on your monitor. For example, if you type INTFACE, you will see INTFACE NEWUSER if you are currently in the NEWUSER mode.

As explained in the “Command Reference” chapter, the DISPLAY command can be used to see current values of selected sub-sets of commands.

Change the Value of a Parameter

To change the current value of parameter(s) for any variable (other than the “immediate” commands, like CONVERS, that are simple actions, not settings):

Step 1. Go to COMMAND mode, as described above (if you are not already there).

Step 2. Type the name of the command for which you want to change the current parameter value(s), type one or more spaces, and enter the new value(s) you want for the parameter(s) for the command.

Step 3. The KPC-3 Plus will immediately change the value(s) to the values(s) you have just entered.

If you enter a value that is not possible (e.g., INTFACE MOON), you will get an error message (“EH?”).

To confirm that the current value(s) is what you want, enter the name of the command and press ENTER. The KPC-3 Plus will send a message consisting of the name of the command and the current value(s) of its parameter(s) for display on your monitor.

Connect to Your Mailbox

You are now ready to try your first connect.

A good first connect (which also shows you an important part of your KPC-3 Plus), is to simply connect to your own personal mailbox, built in to the KPC-3 Plus:

- This is an internal connect, so your radio does not need to be connected.

Step 1. Go to COMMAND mode (if you are not already there), where you will see the “cmd:” prompt.

Step 2. Type CONNECT call (where “call” is your MYPBBS) and press the ENTER (return) key to enter the command. By default, MYPBBS is the callsign you entered right after the autobaud routine, with “-1" appended to it.

Note: To see the current mailbox callsign, type MYPBBS and press the ENTER (return) key. The callsign shown is the one to use when connecting to your mailbox (PBBS)

Step 3. Your KPC-3 Plus will connect you to the mailbox. Your screen should look something like this:

```
cmd: CONNECT NØKN-1
cmd:*** CONNECTED to NØKN-1
[KPC3P-9.1-HM$]
47500 BYTES AVAILABLE IN 15 BLOCK(S)
ENTER COMMAND: B,J,K,L,R,S, or Help>
```

- The commands now shown as available are just for the mailbox. The KPC-3 Plus’ NEWUSER (and TERMINAL) commands become available again when you exit the mailbox.

Step 4. Now you can give any of the mailbox commands shown (B,J,K,L,R,S, or Help). Later you will want to read the “PBBS (Personal Mailbox) section of the “Modes of Operation” chapter for more details. If you now type the HELP command and press return, you’ll see the KPC-3 Plus mailbox help file and then the standard mailbox prompt:

```
ENTER COMMAND: B,J,K,L,R,S, or Help> .
```

Step 5. Now disconnect (just like you will disconnect from any other user).

First, return to command mode by typing a <Ctrl+C> on your keyboard. To do this, hold down the control key (usually marked Ctrl) and type the letter C. Then let go of the control key.

You should now see the “cmd:” prompt.

Next type the DISCONNE command and press return.

You will receive the message *** DISCONNECTED and a new command prompt.

Congratulations. You have now completed your first connect.

Monitor Communications From Nearby Stations

A good way to start using your packet radio station is to monitor communications from nearby stations.

- Amateur packet-radio activity in the US is typically on VHF, on 2 meters, with 145.010 MHz being the most common packet-radio frequency. The most common baud is 1200.

To monitor communications from nearby stations (after you have tuned in to an appropriate frequency):

Step 1. Set MONITOR to ON (if it is not already ON).

Step 2. Disconnect from any other station (or mailbox).

Step 3. As packets are received, you will see them displayed on your screen. A typical received packet might look like this:

```
NØKN>KBØNYK:
Hi Mike. How are you today?
```

The first line is the header line, and shows the callsigns of the stations talking and the following line(s) are the actual message that was sent. In the example, NØKN is the transmitting station, and KBØNYK is the receiving station. You may see the RCV LED light at times, indicating a signal is being received, but nothing displays on your screen. This is normal and may be the result of the settings of the monitoring commands, which are discussed in detail in the KPC-3 Plus Command Reference.

As you monitor, you will begin to learn about other packet stations in your area and then you will probably want to try to connect to one of them.

Communicate Directly with a Nearby Station

Now you are ready to carry on a two-way conversation with another station. Here we will see exactly what you need to do to connect to and converse with another station.

Let’s say that you decide to connect to KBØNYK. The steps needed are as follows:

Step 1. First, be sure you have the KPC-3 Plus in command mode. To do this, type <Ctrl+C> and then press return. You should see a command prompt (cmd:).

Step 2. Now you use the CONNECT command. To connect to KBØNYK, you would simply type CONNECT KBØNYK and press return, just as you did in the sample connect to your mailbox.

- In some cases you may need to use a relay station (called a digipeater) to contact another station. This is similar to using a voice repeater when the station you want to talk to is out of simplex range. To connect to a station using digipeaters, you still use the CONNECT command, but you must also specify the callsigns (or aliases) of the digipeaters that must be used. For instance, if you need to use two digipeaters with callsigns of WØXI and NØGRG in order to connect to KBØNYK, you would give the connect command as “CONNECT KBØNYK VIA WØXI,NØGRG”

Step 3. Once the connection has been made, you'll receive a message back on your screen from the KPC-3 Plus that says *** CONNECTED to KBØNYK (or, if you are using relay stations, *** CONNECTED to KBØNYK via WØXI, NØGRG). At that point you are ready to start talking to the other station since the KPC-3 Plus has already automatically switched into the CONVERS mode.

Step 4. After you have completed your conversation, simply press <Ctrl+C> to return your KPC-3 Plus to the command mode ("cmd:" will be displayed) and then you can instruct your KPC-3 Plus to disconnect as you did above with the mailbox, by entering D (or the full form, DISCONNE) and return.

As you begin working with your KPC-3 Plus packet radio station, other topics that you may wish to explore (with the help of other sections of this manual) include:

- doing hard and soft resets,
- communicating via relays with a distant station,
- using network nodes, and
- working with a GPS device.

This concludes a quick tour of basic uses of your KPC-3 Plus. With just these basics, you can do a lot, but you have a great deal more power if you want to explore the full possibilities.

For more information on all the topics we have covered here, and more, see the following chapters in this manual.

Modes of Operation

This chapter covers the major ways in which you may use your Kantronics TNC. For details on particular commands, see the “Command Reference” chapter.

Packet Mode of Operation

This section adds to the information presented earlier (in the Getting Started and other sections) on how packet radio uses digital information for communication. By familiarizing yourself with the concepts presented here, you’ll be aware of and, if you wish, able to change the default value(s) of the command parameters in your TNC to optimize your packet activities.

Introduction

Information is Organized into “Packets”

A packet is a group of characters with a flag and header at the beginning and a checksum and flag at the end. A flag is a specific character used to signify the beginning and ending of a packet. The header is information indicating who the packet is from, who it is to, any relay stations needed to get to the destination and some control information. A checksum is a complicated mathematical formula that produces a number based on the combination of characters that are in the packet. This number is recalculated by every station that receives the packet, and if it does not match the number that is in the packet, the packet is thrown away, thus near error-free communications. A packet is also called a frame.

Your Packet Unit is a Terminal Node Controller (TNC)

Packet radio modems, or packet modems, are generally referred to as TNCs. This “label” or “moniker” was adopted when the Tucson Area Packet Radio Group (TAPR) developed their first “TNC-1" packet radio modem kit in the early 1980s. The TNC is the workhorse of packet radio. As a listening device it hears an audio signal from the radio, changes the data to digital form, determines if the data is a good packet and sends it to whatever device is attached, usually a computer. As a relay device it also checks the packets it receives and determines if the packets need to be resent, then does so if appropriate. As a sending device it receives digital data from the computer, packetizes it and changes it into audio tones, which are sent out to the radio. The rule the TNC uses to do all of this is called a protocol.

Protocol for Amateur Packet Radio: AX.25

The most commonly used protocol in amateur packet radio is AX.25. The details of the inner workings can be found in a book titled “AX.25 Amateur Packet-Radio Link-Layer Protocol”, which is available from the ARRL. Most TNC users will not need to or care to go into the details. The TNC takes care of the nitty gritty work for you, although there are parameters you can set that determine how efficiently some of that work is done.

Command Mode

In order to change parameters, or give any other instructions to the TNC, you must be in Command Mode. This is the mode you will be in when you first turn on the TNC or after resetting it.

Once you have left Command Mode for any reason, there is a parameter called COMMAND that determines what special character you will use to return to Command Mode. This comes defaulted as a <Ctrl+C> (i.e., while holding down the control key, press “C”, then release both keys).

Whenever you enter Command Mode the TNC will send a prompt to your screen that looks like this:

cmd:

Connected vs. Unproto

There are two ways to send data in packet radio, connected or unproto (unconnected). In the Connected Mode you first establish a connection. Then your TNC will send packets to that specific station and expects acknowledgments in return. If an acknowledgment is not received, the TNC will resend the data (depending on the setting of AX25L2V2, it may send a poll first). The RETRY parameter will determine how many times this is done before the connection is lost due to bad conditions. If the acknowledgment is received, the TNC is happy and will send more data when available. Therefore the Connected Mode, barring impossible conditions, assures that the station you are connected to will receive everything you say, and in the order you say it.

In the Unproto Mode, when your TNC sends a packet, no acknowledgment is expected and no retries are attempted. This mode is often used for calling CQ, for transmitting beacons, and for informal round table chats.

Monitoring and Calling CQ

If you turn the MONITOR command ON, you will see other people’s packets on your screen. You will notice two callsigns at the beginning of each packet separated by a “>”. The first callsign is the station the packet is from, and the second callsign is the station the packet is to. An Unproto packet may have a name or CQ for the second callsign.

To set what will be seen as the “to” callsign for Unproto packets you send, you use the UNPROTO command. This comes defaulted as CQ.

In order to call CQ you must get into the Convers Mode, so that what you are typing to the TNC will be interpreted as data to be sent out on the air and not as commands. To do this type:

K<CR>

Now anything you type will be packetized and sent out on the air. Remember to get back to Command Mode you enter a <Ctrl+C> (default) by holding down the Control key

while pressing “C”. You will be going between Command and Convers Modes depending on whether you want to talk to the TNC or have the TNC packetize what you type to go out on the air.

A Simple Connect

Once you see a station you would like to connect to, be sure you are in Command Mode, and issue a connect request. For example:

C callsign<CR>

where ‘C’ is short for CONNECT and ‘callsign’ is the callsign of the station you wish to connect to.

If for any reason the connection fails, the TNC will send the following message to your screen:

*** RETRY COUNT EXCEEDED
*** DISCONNECTED

When your TNC does receive an acknowledgment for a connect packet, it will display a message on your screen like:

*** CONNECTED TO callsign

Then your TNC will change to the Convers Mode (dependent on the setting of NOMODE and CONMODE). Now what you type will be interpreted by the TNC as data to be sent to the other station and not commands to the TNC. The MCON parameter comes defaulted to OFF. Therefore once you are connected all you will see is what you type and what the person connected to you sends you. Any packets sent by other people will not be monitored unless they are also connected to you on another stream.

Two things determine when the data you send will be packetized. One is the parameter SENDPAC. This is defaulted as the return or ENTER key. So as you are typing your message, whenever you hit the return or ENTER key, you are telling the TNC to make a new packet. A second parameter, PACLEN, determines the maximum length of any packet. If you enter data longer than this length, a packet will be made even though you have not pressed the return or ENTER key.

When you have finished your conversation you need to end the connection. To do this you go into the Command Mode and type a “d” for Disconnect. Remember to press the return or enter key after any command to the TNC. Once your station has received the acknowledgment for the disconnect packet the TNC will send this message to your screen:

*** DISCONNECTED

Either station can issue the disconnect command, no matter which station originated the connect.

Digipeating

Everything we have done so far will only be heard by those within range to hear your signal. With packet radio it is possible to go farther than that. The DIGIPEAT parameter in the TNC comes defaulted ON. This makes your TNC a possible relay station, or digital repeater —digipeater, or just digi for short. In many VHF communities one or more of these is put up in a good, high location and referred to as a dedicated digi. The TNC and radio is all that is needed for the digital repeater to do its job. A computer would be needed if you wanted to change a parameter, but it would not need to stay there for the digi to work. The higher the antenna, the more effective a digi will be, but remember, every TNC has the capability of being a digipeater.

If we turn the MRPT command ON we will begin to see more than just the “from” and “to” stations of the monitored packets. We will also see the callsigns of those stations that have been used as digipeaters. This list of stations is often called a path. Here is an example of what you might see:

```
NØKN>KBØNYK, IAH*,LAG,AUS:
Hi there
```

In this example, NØKN is talking to KBØNYK, using the digipeaters IAH, LAG and AUS. The asterisk beside IAH tells you that you are hearing that digipeater. You will notice that IAH, LAG and AUS are not real callsigns. The TNC provides a parameter (MYALIAS) to set up an alias, which is often easier to remember than a callsign. To make this connection NØKN would have typed the following command to his TNC:

```
C KBØNYK V IAH,LAG,AUS
```

V is short for via and up to 8 digis may be used. You must specify digis in the order they will be encountered along the path from your station to the station you wish to connect to. A space must be typed after the “C” and on both sides of the “V”, but digis are separated by commas. A path can also be used with the Unproto command:

```
U CQ V NOM,LCH,SLI,BIX
```

Unproto sets up the path for anything that is subsequently typed in the Convers Mode where no connection exists. CONNECT issues a connect request to the specified station, via the specified path. Then a virtually error-free conversation can take place between them.

When digipeating, the packet goes all the way from the first station, through all relay stations, to the destination station. Then the response also has to take this same path in reverse. Chances for collisions and, therefore, for retries are multiplied with every digi used. This is often called end-to-end acknowledgment. Another way to get from one place to another is to connect to a “node”. A node will take care of the acknowledgment between it and the next node or end user. See the KA-Node section for more information. Ask your local packeteers about other types of nodes which may be operational in your area, such as K-Net, TheNet, NET/ROM, G8BPQ, and ROSE.

- Kantronics' TNCs support additional, advanced UI digipeating commands. For a description of Kantronics' UI digipeating commands, see "Advanced GPS (or APRS) digipeating", in the GPS Mode of Operation section of this chapter.

Gateways

In Kantronics' multi-port devices (e.g., Kantronics' KAM Plus, KAM XL, and KPC-9612 Plus), a Gateway is also available.

Using a gateway is similar to digipeating except that the retransmission of the packet takes place on another radio port of the TNC other than where it was received. In other words, if you are on VHF and send a packet via a Gateway callsign of a KAM, your packet would be retransmitted on HF on whatever frequency the radio connected to the KAM is set for. When you use one of these gateways on HF, remember that on HF the baud is slower so you may want to lengthen some of your timing parameters. See the KA-Node section for another way to accomplish this with a more reliable connection.

Multi-Connects

The TNC makes it possible for you to talk to more than one person at the same time. Single port TNCs such as the KPC-3 Plus support 26 streams on the one port.

- Multi-port TNCs (e.g., KPC-9612 Plus) support 26 streams per port.

The command MAXUSERS determines how many streams may be used at one time, per port, and the command USERS determines how many people can connect to the TNC per port. An incoming connect uses the next available stream. If the number of streams set by USERS is full, then a station attempting a connect with your TNC will receive a busy message instead of a connect. However, if MAXUSERS is set larger than USERS, you can still issue outgoing connects on additional streams.

To determine which port you are on, simply use the STATUS command, typing STAT at the command prompt. The TNC will report which streams are active and which one you are on. If you wish to remain on the current stream to communicate, no action is necessary. To change streams (to make another connect or to send data to another station already connected to you) type the STREAMSW character, the number of the port (if you are changing ports on a multi-port TNC), and the letter designation of the stream you wish to be on. No return or enter key is necessary.

For example, let's assume you are using a KPC-9612 Plus and you are connected to WØXI on stream A of port 2 but you wish to return to a discussion with NØGRG on stream B of port 1. Whether in command mode or convers mode, simply enter "|1b" and the TNC will switch to port 1, stream B. When you do this, the CON and STA lights on the front panel will switch with you, reporting the status of the new stream.

If you are connected and have MONITOR or MCON OFF, the normal headers containing the "to" and "from" callsigns will not be shown. The setting of STREAMEV will then determine how often you see the stream designator. This parameter comes defaulted OFF, so the stream designators are only shown when a change in streams

occurs. Turning this command ON will make the stream designators show on every connected packet received. Turning STREAMCA ON will also add the callsign of the "from" station beside the stream designator.

Round Table Discussions

Several people talking together present a difficult situation for packet radio since the protocol requires two stations to connect in order to make sure they receive each other's packets. If you wanted to be absolutely sure that everyone got everything you said, you would have to connect to each person and retype everything to each person. That could get a bit cumbersome, so most people use the Unproto Mode and are aware that a collision may occur once in a while. You can usually tell by the conversation if something was missed; if you don't get an answer to a question it's probably not that he is ignoring you, but either the question or the answer was corrupted by a collision.

Selective Monitoring

With MONITOR ON, the BUDLIST command can help in setting up your monitoring so you will see only those packets you want to see. You can list up to 10 calls in BUDLIST.

Timing

Dwait vs. Persistence and Slottime

When the TNC acts as a digipeater, packets received that need to be relayed are retransmitted as soon as the frequency is clear. Other TNCs on frequency (should) hold off or wait their turn, using a PERSISTENCE/SLOTTIME algorithm or DWAIT method to determine when to transmit. This practice has been adopted to reduce collisions with packets being digipeated, since these packets must (by the AX.25 protocol) be acknowledged end-to-end. Kantronics' current TNCs (e.g. KPC-3, KPC-3 Plus, KPC-9612, KPC-9612 Plus, among others), use the PERSIST/SLOTTIME algorithm to gain channel access. The default value for DWAIT is set to zero. If the DWAIT method is used, the TNC sending non-digipeat packets will wait for DWAIT (times 10 milliseconds) time of clear channel before beginning to key-up the radio to transmit.

The algorithm used with the PERSIST and SLOTTIME parameters helps avoid collisions by randomizing the wait time before transmitting. The more random the timing the less chance of two TNCs transmitting at the same time and colliding.

Once the TNC detects a clear frequency it will wait SLOTTIME (times 10 milliseconds). Then, if the channel is still clear, it will generate a random number. If this number is smaller than the setting of PERSIST the TNC will transmit. If it is larger it will wait another SLOTTIME and then generate another random number and again decide whether to transmit or not. When using PERSIST and SLOTTIME you should set DWAIT to 0, since both will be used if specified.

As an example, let's assume that PERSIST is set to 63, and SLOTTIME is set to 10. This value of SLOTTIME results in a random number being generated every 100 milliseconds. When the TNC sees that the channel is clear, it waits 100 ms, then generates a random number between 0 and 255 (inclusive). If, in our example, the

number was 83, then the TNC would not start the key-up of the transmitter since 83 is greater than the 63 PERSIST value. Instead, it would wait an additional 100 ms, and if the channel is still clear, generate a new random number. This time, let's say it comes up with the number 27. Since this is less than the PERSIST value, we now start the key-up of the transmitter to send the packet.

Txdelay

TXDELAY should be adjusted to allow your radio sufficient time to switch from receive mode to transmit mode and develop full power output. If the TNC sends the packet before the radio is at full power, the beginning of the packet will be lost and no one will be able to decode it. TXDELAY also allows the station you are talking to sufficient time to switch from transmit mode back to receive mode. The TNC sends flags during the TXDELAY period, so if someone has this set extra long you will hear a repetitive sound at the beginning of the packet.

Frack (Frame Acknowledgment Time)

If the TNC expects an acknowledgment of a packet it has sent, it will wait FRACK seconds for the acknowledgment. If the acknowledgment is not received it will either send a poll or retransmit the packet, depending on the setting of AX25L2V2. When digis are used, extra time is allowed for each transmission using the following equation:

FRACK * ((2 * n) + 1) seconds

where n is the number of digipeaters. The lower the baud (HBAUD) the higher this parameter should be set, because everything is slower. The length of the transmission (determined by PACLEN and MAXFRAME) also needs to be taken into account when deciding how to set FRACK. Longer packets (and more of them) require more time to be transmitted, more time to be repeated by the digipeater, and so on down the line. The FRACK timer begins when PTT is released (the packet has been sent) and is suspended when data carrier from the radio is present, or when your station is transmitting.

Retries AX.25 Level 2, Version 1 vs. Version 2

The way retries are accomplished depends on AX25L2V2 being OFF or ON. To explain this we will follow a conversation through its path. First let's assume station "A" is connected to station "B" with Version 1 protocol (AX25L2V2 OFF). When station A sends a packet to station B, he expects to receive an acknowledgment back indicating that station B has received the information. In order to verify that the proper packet (or frame) has been acknowledged each frame has a number. This number is sent as a part of the frame so the receiving station knows where this packet belongs in the conversation. The frame numbers range from 0-7 and because of this, we are limited to a MAXFRAME of 7 (we do not want the same frame number reused in the same transmission). This is also true for Version 2. If the first acknowledgment is received, there is really no difference between the two versions, practically speaking. The difference shows up with retries, so let's assume that the packet did not get through on the first attempt.

Let's assume that station A sends frame number 3 to station B. Station B does not receive the frame and therefore no acknowledgment is received by station A. With version 1, the entire packet is retransmitted (with the same frame number) to station B and this continues until station A receives an acknowledgment from station B. This acknowledgment can take two basic forms. The first time station B receives frame 3 he will send an acknowledgment of the form "ready to receive frame 4" <rr4>. If this acknowledgment is sent, and station A did not receive it, station A will again send frame 3. Since station B already received frame 3, he would acknowledge it with the form "I've already got that frame, send me number 4" <rej4>. This is also known as Reject Frame sent. This process would continue until the retry count is exceeded when, under version 1, the sending TNC will initiate a disconnect and discard the packet. (The monitoring of the commands shown in < > depends on the settings of MRESP, MCON and MCOM.)

Now let's look at the same conditions under version 2 (AX25L2V2 ON). Station B does not receive frame number 3 from A and therefore sends no acknowledgment to station A. This time, station A sends a POLL or question to station B saying, in effect, I'm expecting frame number 0 from you; what frame are you expecting from me?" <<RR0>>. Since station B did not receive the frame, station B would respond with <<rr3>>, saying "I'm ready to receive frame 3." At this point, station A, upon receiving the rr3 would immediately resend the entire frame. If station B had already received frame 3 once but the acknowledgment never got to station A the question from station A for the retry would be the same. Station B's response however, would be different. He would respond with "ready to receive frame 4" <<rr4>>. If station A does not receive station B's reply, this "POLL/REPLY" sequence would continue for the number of retries set in the sending TNC; if no response was received, and the RELINK parameter is ON, the TNC at station A would then begin to issue connect requests to station B since there is still an outstanding packet of information. This is the major difference between version 1 and version 2. The connect attempts would then continue for the number of retries set in the TNC, and if no response was received from station B after all of the above, station A would disconnect and discard the packet. The parameter RELINK is defaulted OFF to avoid the reconnect attempt.

Flow Control

The flow control commands insure that the TNC gets everything that is sent to it by the computer and that the computer gets everything the TNC sends it. When the computer sends the TNC data, the TNC stores this data in a buffer until it can packetize it, send it, and get acknowledgments. Similarly, when the TNC sends the computer data, the computer stores the data in a buffer until it can be processed, stored to disk, sent to printer, or whatever.

This buffer area is of limited size; if more data is sent than will fit in the buffer the extra data will be lost. To make sure each device gets all the data it should from the other device, the two devices can tell each other to start and stop sending data. This is called Flow Control and it can be accomplished in either of two ways, via software or via hardware.

Which way you implement this depends on the capabilities of your computer communications program and personal preference. The cable between your computer and TNC must also be wired appropriately.

Software Flow Control

Software flow control sends special characters on the TXD and RXD lines of the RS232 cable. These are the same lines used for sending regular data between the TNC and computer. Software flow control normally sends a <Ctrl+S> to stop data and a <Ctrl+Q> to restart data. When a buffer gets close to full, the device will send a <Ctrl+S> and expect the other device to stop. When the buffer gets nearly empty, it will send a <Ctrl+Q> to tell the other device to send more data. How full or empty a buffer is when the special characters are sent is determined by the program. But, since the regular data lines are being used, a <Ctrl+S> sent as data from the keyboard will also stop data from the TNC. And likewise, if there is a <Ctrl+S> in a file being sent, data flow from the TNC will stop until a <Ctrl+Q> is received.

XFLOW needs to be turned ON for the TNC to use software flow control. XOFF determines the character sent by the TNC to stop the flow of data from the computer, and the XON character restarts the flow. The TNC expects the computer to send the STOP character to stop data to the computer and the START character to restart data. To use software flow control these commands would be set as follows: XFLOW ON, XOFF $13, XON $11, STOP $13, START $11. (Default settings.)

Two commands are provided to make it possible to send and/or receive these special characters in Transparent Mode and still use software flow control. TXFLOW enables flow control sent by the TNC to the computer and TRFLOW enables the TNC to recognize flow control sent by the computer. If both these commands are ON (and the above commands are set as stated) then software flow control will take place in both directions, to and from the TNC and computer.

- If you are in Transparent Mode sending a file, the computer does not need to tell the TNC to stop and start (since you are sending the file). By turning TRFLOW OFF and TXFLOW ON, the TNC will accept START and STOP characters embedded in the file and send them along to the radio as data. Meanwhile, the TNC can send the XOFF and XON characters to the computer to control the flow of data from the computer. When receiving a file, turn TRFLOW ON and TXFLOW OFF; the computer will accept all data and use START and STOP characters to control the flow of data from the TNC.

Hardware Flow Control

Hardware flow control monitors the voltages on the RTS and CTS pins of the RS232 cable. Therefore these two pins must be wired between your TNC and a connected computer. The TNC holds CTS high as long as it can receive data. Once its buffer gets full, it pulls this line low. The computer program monitors this line. When it is pulled low, it knows to stop sending data. When the line is again pulled high by the TNC, the computer program will restart sending data. On the other hand the computer holds RTS high as long as it can receive data and pulls it low to tell the TNC to stop sending data. The TNC always uses hardware flow control, so only wire the RTS and CTS pins if your computer program is also using hardware flow control.

- HyperTerminal uses hardware flow control, but some communication software does not.

Convers Mode vs. Transparent Mode

In the Convers Mode there are many special characters. To list a few:

Command	Default	Description
SENDPAC	<Ctrl+M>	Causes a packet to be formed
DELETE	<Ctrl+H>	Backspace character
REDISPLAY	<Ctrl+R>	Redisplays the keyboard buffer
CANLINE	<Ctrl+X>	Cancels a line
STOP	<Ctrl+S>	Stops output from TNC to computer
PASS	<Ctrl+V>	Pass a special character

These characters are all very useful when having a packet conversation with someone. If you want to send a packet you hit the return. If you make a mistake you can erase characters, using the delete or backspace key, or kill the whole line with <Ctrl+X>. And if you really want to transmit one of these characters, you can always precede it with a PASS character.

Transparent Mode was designed for the sending of files, whether they be ASCII data files or program files. Special characters have no meaning to the TNC in transparent mode, they are just characters to be put in a packet and sent to the radio.

In Transparent Mode, a SENDPAC character will not cause data to be formed into packets; instead, packets are formed at a rate that is set by PACTIME.

All monitor commands are treated as OFF in Transparent Mode. All you will see is what is being sent to you. You would probably want to set USERS to 1 so no one interferes with the transfer. The setting of ECHO is also ignored. Even if ECHO is ON Transparent Mode will not echo to the attached terminal. Some programs allow for local echoing to the screen while uploading.

Getting Out of Transparent

Getting into the Transparent Mode is easy: you just type a “t” in Command Mode. But since Transparent Mode allows the sending of all characters you can not get out of Transparent Mode by just typing a <Ctrl+C> (COMMAND character) as in Convers Mode. In order to get out of Transparent Mode, you must follow a special sequence, or use a modem break if your program supports one. The special sequence must be followed precisely. This example assumes the COMMAND character is <Ctrl+C> and CMDTIME is 1 second:

- Wait at least 1 second since the last character was sent from the computer to the TNC
- Type a <Ctrl+C>

- Within 1 second type a second <Ctrl+C>
- Within 1 second type a third <Ctrl+C>
- Wait 1 second and the cmd: prompt should appear

If the guard time of one second before and after the three <Ctrl+C>s is not there, the TNC assumes that they are data and sends them to the radio, so be sure to allow at least one second before and after the three <Ctrl+C>s.

Remote Access to Your TNC

You can connect to your TNC from a remote station and change values of your TNC's parameters. This allows you to add or delete stations from the LLIST, change the size of the PBBS, change the MYCALL and so on, all remotely. Extreme caution must be used when you are accessing your TNC from a remote location. There is no built-in safeguard, and as such it is possible for you to change parameters such that the remote TNC will no longer communicate with you.

In order to change parameters in a remote TNC, the RTEXT in the remote TNC must be set to a text string that will be used as the password string. For instance, you might set your RTEXT to:

RTEXT Code

The remote TNC must also have its MYREMOTE set to a unique callsign (i.e. WØABC-4, or XYZREM). If these two parameters are not set, remote access to the command set of this TNC is not possible. When these parameters are set, you can connect to the MYREMOTE callsign of the remote TNC.

When the connection is made, the remote TNC will send three lines of numbers. The numbers may look like:

1 1 1 3 4 3
3 1 4 3 1 3
2 1 1 2 1 3

You must then pick ONE of these lines and decode the password string. Let's say you choose to decode line 3 (2 1 1 2 1 3). Rewriting your RTEXT string to make this easier you would have:

Character #:	1	2	3	4
Letter:	C	o	d	e

Therefore, to gain remote access, you must send the following string:

oCCoCd

(since small letter o is at position 2, capital C is at position 1, and so on)

Note that case is significant and spaces are considered valid characters.

If you fail to properly decode the password, the TNC will send three new lines of numbers. You will be given a maximum of three attempts to properly decode the password string. If you fail in three attempts, the TNC will disconnect you and disable connects to the MYREMOTE for 15 minutes. Also, if you connect to the MYREMOTE and start the password sequence but then disconnect, the penalty timer is in effect for 15 minutes.

Be careful when using the remote access feature. You can change ANY command in the TNC without restriction, but this can lead to problems. For instance, if you change the INTFACE command to KISS and then send a RESET command, the remote TNC will be placed into the KISS mode and will completely quit talking to the radio! Also, if you connect to the MYREMOTE of a TNC and then issue a command like CONNECT W1ABC, the remote TNC will indeed connect to the station, but there will be no data sent to you from that connection. The connected data would be sent to the serial port of that remote TNC. We urge EXTREME caution when using the remote access! Note also that any command that causes a reset (i.e. NUMNODES, PBBS, MAXUSERS) will disconnect all current users (PBBS, NODE, and YOU).

One other minor limitation applies to receiving data from the remote TNC. When you give any command, the response is limited to 300 characters so some commands will not fully display their response. If you give the command DISPLAY, the full list will begin, but you will not receive the entire list because of the limitation on this buffer.

PBBS (Personal Mailbox)

Introduction

Your TNC includes a Personal Bulletin Board System (PBBS), a mailbox, which is capable of storing and forwarding messages for you and other users. This PBBS provides the same message facilities as a computer based packet radio BBS, normally referred to as a full-service BBS, including the forwarding of Bulletins, Private mail, and NTS traffic (for more information on the national network of full-service PC-based packet BBS systems, see a current ARRL operations manual).

Historical note: The first BBSs in packet radio were PC-based, using software developed by Hank Oredson, WØRLI. A national system of packet-radio based BBSs has developed, permitting the automatic forwarding of messages, and files, from one BBS to another. Today, the large capacity packet radio BBSs are still PC-based, but TNC-based BBSs now provide similar facilities, with medium capacities.

The personal mailbox is compatible with the large community bulletin board systems (RLI, MBL, etc) and will allow them to forward mail for you directly into your TNC. You may also place messages in your mailbox, and if the local Community BBS system allows, your TNC mailbox will reverse forward these messages from your personal mailbox into the community system on request. You can even set up your TNC mailbox to automatically connect to the community BBS and forward any messages it has.

Using Your PBBS

In order to use any Kantronics TNC PBBS (even your own), first get the cmd: prompt on your TNC, and then connect to the callsign of the PBBS. For instance, if your MYPBBS is NØKN-1, you would connect to it simply by typing “C NØKN-1". In this example, the PBBS is in your own TNC, so no packets would be transmitted, but you would connect to the PBBS and receive the same prompt as if you had connected to someone else’s PBBS.

When you connect, you will first see the message from your TNC indicating that you are connected:

```
*** CONNECTED to NØKN-1
```

The PBBS will then send you its SID (Status ID - e.g., equipment ID) and initial sign-on message. If a PTEXT is defined, the TNC will send it as the next line, and then send the PBBS command prompt. For example, a Kantronics KPC-3 Plus might respond to your PBBS connect with the following:

```
[KPC3P-9.1-HM$]
475000 BYTES AVAILABLE IN 15 BLOCK(S)
PTEXT would be here (if any)
ENTER COMMAND: B,J,K,L,R,S, or Help >
```

Using anyone else's Kantronics-based PBBS would result in a similar prompt sequence.

At this point you are ready to send a message to another user, read a message to yourself (if any has been received), or issue any other mailbox command.

Let's assume you want to send a message to KBØNYK. You would now use the Send Private command:

```
SP KBØNYK
```

and the TNC responds with:

```
SUBJECT:
```

Now enter a short subject line:

```
Just a quick question
```

The TNC responds with:

```
ENTER MESSAGE n—END WITH CTRL-Z OR /EX ON A SINGLE LINE
```

(where "n" is the next available message number in the mailbox).

Now you enter the text of your message (message n). To end the message and have it saved, type a <Ctrl+Z> (hold down the control key and press Z) or type /EX. The <Ctrl+Z> or /EX must be on a line by itself —do not type anything else on this line. When the message has been ended properly, the PBBS responds with:

```
MESSAGE SAVED
ENTER COMMAND: B,J,K,L,R,S, or Help >
```

You may now enter more mailbox commands.

The "Mail" indicator on the front panel of your TNC blinks to alert you to unread mail addressed to you (i.e., to the callsigns of MYCALL or MYPBBS). See the TNC command PBLIST for how you can add other callsigns to the list of callsigns that are to be included as "mine." The PBLIST is also used with the PBPERSON command and the following PBBS commands: LMINE, KMINE, and RMINE. Note that the mail-status indicator on the front panel of the TNC is turned ON (without blinking) when someone is connected to your PBBS.

PBBS Commands

The commands available to users connecting to your Kantronics TNC PBBS (and you connecting directly) are as follows:

B(ye)

This command is entered by the PBBS user to disconnect from the PBBS.

E(dit) n [BPTYNFH] [>tocall] [<fromcall] [@BBS] "old" "new"

This command is available only to the system operator (SYSOP) or to a user at the keyboard of a computer connected directly to the TNC (i.e., not to a remote connection, except for the SYSOP). For an explanation of this command, see the section below on “Editing Message Headers.”

H(elp)

Displays a HELP menu.

J(heard)

Displays a list of stations recently heard by the TNC. The list will include a date and time stamp indicating when each station was heard.

J(heard) S(hort)

This command is similar to the J(heard) command above, but only displays the callsigns of the stations heard.

J(heard) L(ong)

This command is similar to the J(heard) command above, but also displays any digipeaters used by the stations it has heard and the destination callsign.

L(ist) [x [y]] [;]

Lists all messages in the mailbox which you are allowed to read. This will include all BULLETINS, TRAFFIC, and any PRIVATE messages addressed TO you or sent by you. If you are the SYSOP (keyboard or remote) ALL messages will be listed. You have the option of specifying a beginning message (x = START #) and, if you do so, an ending message for the list (y = END #). Also, specifying the optional semi-colon (;) in any List command will also show the @BBS and BID (BULLETIN ID - see section on “Sending Messages”) of the messages (if any).

L(ist) <|> call [;]

Allows you to list only those messages in the mailbox which are addressed to a specific callsign (>), or which were sent by a specific callsign (<). To list only those messages addressed to AMSAT for instance, you would give the command

```
L >AMSAT.
```

LB [;]

Lists all BULLETINS in the mailbox.

LC [cat [;]]

Using the LC command by itself will cause the PBBS to list the TO field of all BULLETINS in the mailbox. This will just be a list of the actual TO fields, and not a list of the messages. If you specify a category (i.e. LC RACES) the PBBS will list the full message headers of all BULLETINS addressed to that category.

LL n [;]

Lists the most recent n number of messages in the mailbox. Again, only BULLETINS, TRAFFIC, and PRIVATE, which you are allowed to read, will be listed.

LM(ine) [;]

Lists all messages in the mailbox which are addressed to you.

LO [+|-]

This command allows you to change the order in which messages are listed. When set to +, the messages are listed in ascending numerical order (oldest to newest). When this command is set to "-", the messages will be listed starting with the highest message number (newest to oldest). This command will not be available if the SYSOP has the PBLO command in his TNC set to FIXED.

LT [;]

Causes the PBBS to list all TRAFFIC messages that are currently in the PBBS.

K(ill) n

Deletes message number n from the mailbox. You may only delete TRAFFIC messages, PRIVATE messages addressed TO or FROM you, or BULLETINS you sent. The SYSOP may delete any message, including BULLETINS.

KM(ine)

Deletes any messages in the PBBS addressed to you that you have read. If you have not read a message addressed to you, it will not be deleted.

NL n

The NL (number of lines) command controls how many lines are output at a time when a large message is read. After n lines are printed, the message "PRESS Q TO QUIT, C FOR CONTINUOUS, RETURN FOR MORE" will appear. 'Q' will abort the printout of the message. 'C' will cause the rest of the message to be printed without stopping. Pressing RETURN will print the next n lines.

R(ead) n

Reads a specific message by number. Only PRIVATE messages addressed TO you, or sent by you, may be read, as well as any BULLETIN or TRAFFIC messages. After you read a PRIVATE message addressed to you, the STATUS flag will automatically be set to Y—it has been read.

RH n

The RH (Read with headers) command is a variation of the Read command. Using RH instead of R gives you message headers as well as the other information provided by the Read command. The message header contains the date/time the message was received at each forwarding BBS along the path to your station.

RM(ine)

Displays all messages in the PBBS addressed to you that you have not already read.

S(end) call

Sends a PRIVATE message to the callsign specified. Using S(end) is the same as using the SP command.

SB cat

The SB (Send Bulletin) command is used to send a BULLETIN to the PBBS.

SP call

The SP (Send Private) command is used to send a PRIVATE message to a specified callsign. Using this command is the same as using the S(end) command.

ST zip

The ST (Send Traffic) command is used to send NTS type traffic messages to the PBBS. Some of these commands are described in more detail below.

U(sers)

The Users command will display any other users who may also be currently connected to the PBBS.

Sending Messages

The SEND command (and its many forms) allows the following syntax:

S	call	[@ bbcall[.haddr]]	[$ mid]
SP	call	[@ bbcall[.haddr]]	[$ mid]
ST	zip	[@ location[.haddr]]	
SB	cat	[@ location[.haddr]]	[$ bid]

where:

- **call** is the callsign of the station the message is addressed to.
- **bbcall** is the callsign of a full service BBS to deliver the message to.
- **haddr** is the complete Hierarchical address designator for the BBS system or destination of the message (see the section below on “Hierarchical Addresses” for more detailed information). Contact your local community BBS SYSOP for complete information on local Hierarchical addressing.
- **location** is the designator used for distribution of the message. For TRAFFIC, this should be NTSxx where xx is the two letter POSTAL code for the state.
- **mid** is the MESSAGE ID assigned to the message by the originator.
- **bid** is the BULLETIN ID assigned to the message by the originator.
- **zip** is the 5 digit postal zip code (or postal code)
- **cat** is the message category. For instance, a message requesting help on a subject may be sent to the category HELP, info sent to INFO, items for sale to SALE, etc. Contact your local community BBS SYSOP for some other examples and suggestions.

Some examples of commands would be:

```
SP WB5BBW @ W5AC.#STX.TX.USA.NOAM
```

This command sends a private message to WB5BBW. The message should be sent to the W5AC BBS system, in South Texas (.#STX), which is in Texas (.TX), which is in the USA (.USA), which is in North America (.NOAM) where WB5BBW can retrieve it.

```
ST 88030 @ NTSNM
```

This command sends an NTS traffic message to a non-ham, or to someone who is not on packet, living in zip code 88030 which is in New Mexico. The location field contains the NTSxx (xx = NM) to indicate that the 88030 zip code is in New Mexico.

```
SB RACES @ ALLUS $RACESBUL.010
```

This command sends a bulletin addressed to RACES, which should be sent to all BBS system in the USA (ALLUS) and has been assigned the Bulletin ID (BID) RACESBUL.010. This BID prevents the same message from being duplicated as it travels throughout the BBS system.

When you send a message to the PBBS, you must include the @BBS (bbcall[.haddr]) field if you want the message to be reverse forwarded from the PBBS to a full-service BBS system. Any message entered into the PBBS over the radio will initially be marked with a status of H (held) and will not be reverse forwarded until the SYSOP has edited the message header and changed the H flag. This gives the SYSOP full control over the messages relayed by his station.

Messages entered from the local keyboard connected to the TNC do not require editing in order to be reverse forwarded, but must include at least an @BBS field to enable the reverse forwarding.

Listing Messages

When you list messages with any of the LIST commands, you will get a display similar to the one shown below. If you specify the optional semi-colon (;) you will also see the @BBS field (if it exists) in square braces after each line.

```
MSG#  ST    SIZE  TO     FROM     DATE                SUBJECT
6     B     45    KEPS   W3IWI    12/19/91 09:37:11   2 Line Element set
4     B     26    HELP   WB5BBW   12/19/91 09:34:05   Xerox 820
3     T     38    66044  WØOUU    12/19/91 09:33:42   QTC Lawrence 913/842
2     PN    14    NØAPJ  WØSC     12/19/91 09:33:27   AMTOR
1     B     30    ALL    NØKN     12/19/91 09:32:49   Need help on AMTOR
9712 BYTES AVAILABLE
NEXT MESSAGE NUMBER 7
ENTER COMMAND: B,J,K,L,R,S, or Help
```

The message number (MSG#) is listed, followed by the STATUS of the message. This status includes the message type (B=Bulletin, T=NTS traffic, and P=Personal message). The second character in the ST column is the current status of the message.

A **Bulletin status** (type B) can be:

F—it has already been forwarded to another full-service BBS

H—it is being held for review by the SYSOP because it was entered into the PBBS over the radio.

An **NTS traffic message** (type T) may have a status of:

H—indicating that it is being held for review by the SYSOP before it may be forwarded.

The **Private message** (type P) can have the following status characters in the second position:

H—This is a personal message that has an @BBS field but is being held for review by the SYSOP before it may be forwarded.

N—This message is a Personal message that has not been forwarded and has not been read by the station it is addressed to. If it is forwarded to a full-service BBS, it will be deleted if PBKILLFW is ON.

Y—This message has been read by the station it is addressed to, but has not been killed. It will not be forwarded even if it has an @BBS since it has already been read.

Reading Messages

To read a message (e.g., a bulletin such as that in the previous section listed as being to ALL), use the READ command (see PBBS command section for details on using this command).

Editing Message Headers

The following command can be used to edit message headers:

E(dit) n [BPTYNFH] [>tocall] [<fromcall] [@BBS] “old” “new”

This command, available only to the system operator (SYSOP), or to a user at the keyboard, allows the sysop to edit the message headers listed below for any message in the mailbox. You are the sysop if you can connect to the PBBS from the RS232 port. You can also gain sysop status by connecting to the PBBS remotely (by radio); see the section titled “Remote Sysop Access to your PBBS” for more detail.

You can change:

- The TYPE of the message:
  - B — Bulletin
  - P — Private
  - T — Traffic
- The STATUS of the message:
  - Y — Yes [it has been read]
  - N — No [it has not been read]
  - F — Forwarded [it has been forwarded]
  - H — Held [it is not available for reverse forwarding]

    Note: F and H are toggles. Entering F switches the status of the message from F to NOTF or from NOTF to F. Similarly, entering H switches the status of the message from H to NOT H or from NOT H to H.
- Who the message is to or from:
  - >tocall
  - <fromcall
- The destination mailbox (@BBS):
  - @BBS[.haddr]

- And the subject or text of the message

When a message is listed, the “tocall” and “fromcall” appear in the TO and FROM columns, respectively. If a message has been entered with an “@BBS” for forwarding, the complete hierarchical address is shown when the message is read:

```
MSG2 02/10/92 10:30:58 FROM KBØNYK TO HELP
@wa4ewv.#stx.tx.usa.noam
```

The @BBS is also listed when using the semi-colon (;) option with any of the list commands.

You can access the Edit command by connecting to the mailbox from the attached terminal, or by connecting over the radio. If you connect over the radio, you must gain SYSOP privilege by giving the SYSOP command. When you give the SYSOP command, the PBBS will send you a password verification string, which must be properly responded to in order to gain SYSOP access. This is explained in detail later.

Let’s say you want to edit message number 2 which currently is a PRIVATE message addressed to WØXI. The message has been read by WØXI so it shows a status of Y. It may list as:

MSG#	ST	SIZE	TO	FROM	DATE	SUBJECT
2	PY	53	WØXI	NØKN	01/14/92 03:36:45	Good afternoon

Let’s say you want to change this so that it is addressed to NØGRG and also change the Y flag to N to show that it hasn’t been read. To do this, you connect to your PBBS (either from the KEYBOARD or as SYSOP over the radio), and issue the command:

```
e 2 N >NØGRG
```

You could do this with two separate commands, or it may be accomplished with the single command shown above.

You can change a subject or text in the message by entering two strings in the Edit command. The *first* occurrence of the string “old” will be replaced by the string “new”. For example, to change the subject from Good afternoon to Good morning:

```
e 2 “afternoon” “morning”
```

Hierarchical Addresses

Hierarchical addresses (explained below) are used to accommodate the national packet traffic system, including messages processed by PC-based packet radio BBSs.

The HTEXT command allows you to set your Hierarchical address. This is used whenever your PBBS forwards mail to another BBS to provide complete return addressing capability. Other PBBS features include the ability to select whether or not to kill private and traffic messages after they have been forwarded, and whether or not to HOLD messages received over the radio.

A hierarchical address consists of your state, country, and continent codes, separated by periods. Some areas (in large states particularly) use sub-state designators. For instance, a station in Rhode Island might use a hierarchical address of RI.USA.NOAM. This means Rhode Island (RI) which is in the United States (USA) which is in North America (NOAM). In this case, you should set the HTEXT to RI.USA.NOAM. If you are unsure of your hierarchical address, contact your local packet BBS sysop and ask what the proper addressing is for your location.

The PHEARD command will show you the callsigns of stations recently connecting to your mailbox along with their start and stop times.

Advanced Configuration of Your PBBS

When you first enter your callsign into the TNC, your PBBS will automatically be enabled. The MYPBBS callsign is set to your basic call with an SSID of -1, and the PBBS is allocated 100 k bytes of RAM (assuming you have the standard 128 k byte RAM chip installed).

You may change the size of the PBBS using the PBBS command. The maximum amount of memory you can allocate will depend on the amount of free memory available. NUMNODES, MAXUSERS, and MYREMOTE, etc., will affect the amount of available memory.

If you change the size of the mailbox, the TNC will not renumber any existing messages, and if the new size is large enough for all existing messages, no messages will be lost. If you want to renumber the messages (starting with 1) give the PBBS n command with n being the current size (i.e., n kbytes of memory assigned to the PBBS).

- To see the current size of the mailbox, enter the command “PBBS” (without any number) at the Command prompt.

You can also change the maximum number of users who can connect to the PBBS at the same time. PBUSERS will cause a reset if it is changed.

At times, you may be away from your computer and would like to switch a user into your mailbox automatically if he connects to your MYCALL. This can be accomplished by setting the CMSG command to PBBS. When this is done, a user who connects to your MYCALL will be sent your CTEXT (if any) and then be automatically connected to the PBBS. The TNC will then send the PBBS System ID (SID) and sign on message. The SID is enclosed in square brackets and consists of the unit name, firmware version, and the supported feature set.

For example the Kantronics KPC-3 Plus SID is:

[KPC3P-9.1-HM$]

This is the unit name (KPC3P), version number (9.0) and the feature set (HM$). The H means it supports Hierarchical forwarding, the M stands for Message ID, and the $ indicates BID support. These identifiers are the SID definitions published by Hank Oredson (WØRLI) with his Community BBS system.

You can customize a greeting message to be sent to a user who connects to your PBBS by using the PTEXT command. This command accepts up to 128 characters as a text string to be sent to the user immediately after the SID is sent.

If it becomes necessary to disconnect a station from your PBBS, you can use the local terminal connected to your TNC to accomplish this (see DISCONNECT MYPBBS). If a station connects to your PBBS and no activity occurs on the connection for 15 minutes, the PBBS will automatically disconnect the user in order to make your PBBS available to others.

Note: To get forwarding or reverse forwarding, you need to have HTEXT set. HTEXT specifies your Hierarchical address, which is needed for complete return addressing capability.

If a community bulletin board forwards messages into your PBBS, it sends you many lines beginning with “R:”. These are routing headers that show the complete path taken by this message. By default, these headers will be stored in your PBBS with the message. If you choose, you may prevent your PBBS from storing these by setting the PBHEADER command OFF (for more information, see the section below titled “Routing Lines”).

If you want your PBBS to only accept messages for you, you can set the PBPERSON command ON. When set ON, this command will make your PBBS only accept messages that are addressed to your MYCALL, MYPBBS or PBLIST.

Finally, the PBLO command is used to determine the order in which messages are listed to a user. When PBLO is set to OLD, messages will be sent oldest first (message 1, then 2, etc). When set to NEW, the most recent message will be listed first. The second parameter of this command determines whether or not you will allow a PBBS user to change the listing order while he is connected. When set to FIXED, the user cannot change the order, and when set to VARIABLE, the user may change the order by connecting to the PBBS and using the LO command.

Remote SYSOP Access to the PBBS

You can gain remote SYSOP access to your PBBS (e.g., to edit messages, using the EDIT command) by first connecting to your PBBS and then entering the SYSOP command. When you enter the SYSOP command, you will receive three sets of numbers from the PBBS. These numbers indicate the character positions of the RTEXT to be used as the password for logging on for this remote access.

For instance, let’s say your RTEXT is:

> This is a sample rtext.

Now when you give the SYSOP command, the TNC might respond with:

1	12	3	18	6	9
2	10	22	5	7	18
13	16	4	9	1	20

If you choose the first set of numbers, you should send the following as a response:

Tairia

(T is the 1st letter, a is the 12th letter, i is the 3rd letter, and so on. See the RTEXT command in the Commands section for a more detailed explanation.)

NOTE: Spaces DO count as characters, and case is significant!

Reverse Forwarding Messages from Your Mailbox

The TNC mailbox will allow you to enter messages, which will be forwarded by full-service BBSs (RLI, MBL, etc). These messages have a special format, and can be entered in any personal mailbox. Let's suppose you want to send a message to WA4EWV who lives in Texas. You know his home BBS is WB5BBW, so you can put this message in the PBBS with the command:

S WA4EWV @ WB5BBW

Entering an @ BBS will cause the TNC to reverse forward this message to a full service BBS when requested by the full service BBS. In order to improve the chances of this message reaching its destination, you should always enter the message with complete hierarchical forwarding:

S WA4EWV @ WB5BBW.#STX.TX.USA.NOAM

Complete information on Hierarchical forwarding can be obtained from your local BBS system operator, but basically the first field after the @ symbol is the HOME BBS of the station you are trying to send a message. The next several fields (separated by periods) are the state (two-letter postal abbreviation), country, and continent. In this case, since Texas is so large, it is sub-divided into smaller areas. These are indicated with the # symbol (in this case #STX—South Texas).

Messages entered into your mailbox in this format will be reverse forwarded to the full service BBS when requested, and the following rules apply:

The TNC acts like a "smart BBS" when forwarding to or from a full service BBS. This means that it will no longer send the SUBJECT: prompt, nor will it send the ENTER MESSAGE prompt. You will also notice that when a full-service BBS connects to your PBBS, the TNC does not send the usual ENTER COMMAND prompt, but only the > is sent. This is designed to reduce the amount of data on the packet network, since "smart" BBSs know what is expected of them.

Once a Private or Traffic message has been successfully forwarded out of your mailbox, it will be deleted from the PBBS if PBKILLFW is ON. Bulletins will be marked with a status of "F" and will remain in the PBBS.

Selecting a Home Full-Service BBS.

One important use of your Kantronics PBBS is to connect to the national system of packet-radio based BBSs, which was described earlier in the "historical note" at the

beginning of the PBBS section. You can use a “home” full service BBS for sending and receiving mail, much as you would use a P.O. Box at a postal facility. The full-service BBS should always be on and available to handle mail for you.

Each PBBS user should select one (and only one) PC-based, full-service BBS for sending and receiving mail. This BBS is then called your HOME BBS and should not be changed unless you move to a new location. When you connect to your home bulletin board system and list the messages (using the L command), you will see a list containing information about each message on the system. A list of messages on one local full-service system might look like this:

Msg #	TSL	Size	To	From	@ BBS	Date/Time	Subject
======	===	====	=====	======	=======	====/====	=======
59765	B$	1491	NASA	N5IST	ALLUS	1004/1529	GALILEO STATUS 09/30/93
59764	BNL	468	WX	N0NEJ		1012/1017	KC Forecast 10/12 400am
59763	BNL	659	WX	N0NEJ		1012/1017	MO Forecast 10/12 400am
59759	B$	2240	NASA	N5IST	ALLUS	1004/1529	MARS OBS. STATUS 9/27/93
59758	B$	1642	NASA	N5IST	ALLUS	1004/1529	MARS OBS. STATUS 9/22/93

This list shows the message number, type and status information, the size of the message, the addressee (TO field), the originator (FROM field), and the @BBS field. In addition the list shows the date and time the message was received at this BBS and a short subject for the message.

Routing Lines

Under current FCC requirements, BBS systems that can store and forward messages without an operator being present must have the capability to provide a record of the path the message has taken from its origination. To accomplish this, BBSs include a routing line, beginning with “R:”. This “R:” line includes the date and time the message was received, message number, BBS call and hierarchical routing information.

When you read a message using the R command (e.g. R 59765) you see the header displayed. For example:

```
From        : N5IST
To          : NASA @ALLUS
Type/status : B$
Date/time   : 04-Oct 15:29
Bid         : NASA0930.GAL
Message #   : 59765
Title       : GALILEO STATUS 09/30/93
Path        : !NØKN!N0LLY!N0OER!N0OBM!NX0R!AG0N
              !N7MMC!KT0H!KA0WIN!N5IST!
```

The Path: statement in the header lists the most recent BBS systems that have been used to relay this message from its origin to the BBS you read it from. This path information is required by the FCC to allow them complete traceability for any message in the system. What you see in the PATH statement is not the complete information on

the routing, but simply a summary of the systems that have handled the message. To see the complete information, BBS systems allow a second version of the READ

command (RH or V) that will display more routing information. A routing list from a bulletin appears below.

R:931012/1107 27268@NØKN.#NEKS.KS.USA.NOAM
R:931012/1025 16433@N0LLY.#NEKS.KS.USA.NOAM
R:931011/2021 928@N0OER.#NEKS.KS.USA.NOAM
R:931008/1814 20728@N0OBM.#NCKS.KS.USA.NOAM
R:931008/2003 19520@NX0R.#NKS.KS.USA.NOAM
R:931008/1153 30798@AG0N.#WNE.NE.USA.NOAM
R:931007/1147 35850@N7MMC.#SEWY.WY.USA.NOAM
R:931007/1712 49403@KT0H.#NECO.CO.USA.NOAM
R:931007/1639 63792@KA0WIN.#SECO.CO.USA.NOAM
R:931004/1529 46383@N5IST.#WTX.TX.USA.NOAM

By examining this list from the bottom up, we may see that the message entered the system on October 4, 1993 at 15:29 (R:931004/1529). It was message number 46383 on the N5IST BBS (@N5IST) which is located in West Texas (#WTX), which is in Texas (TX), which is in the United States (USA) which is in North America (NOAM). From this station, it was relayed on October 7 at 16:39 to the KA0WIN BBS in Southeastern Colorado. By following this information it is possible to determine where the message traveled and when it was relayed from each station. The information following the @BBS callsign is called the **hierarchical routing information** (in this case #WTX.TX.USA.NOAM).

When you connect to your local BBS and send a message, that BBS automatically generates this R: line. As the message is sent to its destination, each BBS adds its own R: line to the message. Besides the requirement of the FCC, the R: line provides a method for any user, anywhere in the world, to send a reply or respond to your message. As the message is passed through the many BBSs, each BBS will add you into its White Pages —a directory of packet users. Each BBS makes note that you (the originator of the message) sent the message, and that you entered the message at the BBS listed in the last R: line in the message.

Because of this, a distant user can simply send a reply using the send reply (SR) command of his local BBS. That BBS will then address the message to you using the @BBS and hierarchical routing information in the last R: line of the message you sent. A user may also simply use the send private command (SP) to send a message to you. If the user does not enter complete addressing on his SP command, the BBS will attempt to look up your call in its White Pages and add the routing automatically. However, if the user supplied complete addressing information, the BBS would normally assume it is correct and not check the White Pages.

BBSs use this hierarchical information to send the message back to you. The message someone sent to you (using the above example) would be addressed to URCALL@ N5IST.#WTX.TX.USA.NOAM. As the message passes through the BBS system for forwarding, the BBS first looks at the callsign of the addressee (URCALL). If that BBS doesn't know how to forward the message to you, it then looks at the @BBS field (N5IST). If it doesn't have any information on how to forward to N5IST, it looks at the first part of the hierarchical address (#WTX), not knowing that, it would then look at the next part of the hierarchical address (TX). Assuming this BBS is in the United States, it

knows TX means Texas and knows that this message needs to be relayed to a station in that area.

Once the message reaches the first BBS in Texas, that system must use the previous field for forwarding (#WTX). Once it reaches a system in West Texas, the forwarding occurs based on the @BBS.

When the message reaches the BBS specified in the @BBS field, it can forward the message directly to you, since you are using that system as your HOME BBS.

When you enter a message into your Kantronics PBBS and supply the routing information, that message may be forwarded automatically to another BBS. When the message is forwarded from your Kantronics mailbox, an R: line is included as the originating BBS. This line includes the same information as any other BBS.

This R: line consists of the date/time the message was entered into your PBBS, the message number, your MYCALL (URCALL) and the HTEXT you have set.

For instance, your R: line might be:

R:931008/1255 23@URCALL.#WTX.TX.USA.NOAM

Some BBS operator groups are insisting that your system is NOT a BBS, and therefore should not include R: lines. Their reasoning is that in the above example, EVERY BBS in West Texas would have to know how to send messages to your callsign not just to your HOME BBS. One solution to this is to include the callsign of your HOME BBS as part of your HTEXT. This would change your R: line to:

R:931008/1255 23@URCALL.N5IST.#WTX.TX.USA.NOAM

As this forwards through the system, all West Texas BBSs can still forward the message to N5IST because his call is a part of the hierarchical routing.

As of this writing, there seem to be at least two groups with strong opinions on the use of, or prohibition of, R: lines by TNC based PBBSs. Some think the volunteer BBS network may be overloaded by personal boards including the R: lines; others insist that the R: lines are required by regulations. We suggest you adapt to local custom by turning the R: line feature ON or OFF accordingly.

If your local SYSOP demands that you not add R: lines to your messages, you must set the PBPERSON command ON. This will limit your PBBS to receiving messages addressed ONLY to your MYCALL or your MYPBBS call. In addition, your PBBS will only forward messages from YOU (no third-party messages) and will not add the R: line to the routing.

Hint: When you read messages with the normal READ command, the TNC will not display those R: lines, but instead will display a short PATH: line, indicating the most recent BBSs that have handled this message. If you want to read the full routing, use the RH command to read the message with all the headers displayed.

This concludes the section on PBBS mode of operation.

GPS NMEA Interfacing

Most Kantronics' TNCs, e.g. KPC-3 Plus, KPC-9612 Plus, KAM XL, and others, support the GPS mode, explained below. Each contains firmware to interface with a GPS device supporting the NMEA 0183 interface standard; commands to configure and control unproto (path) retransmissions of the NMEA strings, and provides RAM storage for location messages (strings) for later retrieval.

- Among Kantronics' TNCs supporting GPS, the only difference between devices is the number of radio ports available for transmission of data. For example, the KPC-3 Plus is a single port device and the KPC 9612 Plus is a multi-port device.

Overview

The Global Positioning System (GPS) was developed initially for military use by the U.S. Department of Defense, to provide worldwide positioning and navigational information for U.S. military forces. Since then, a broad range of commercial, civilian, and amateur radio applications have been developed, based on the GPS. The GPS consists of 24 satellites, each of which transmits positioning information continuously. GPS receiving units - devices that contain a receiver, LCD screen, and (usually) a data port - collect this positional information and use it to compute position. This information can then be retransmitted by any means possible to report the position of the GPS unit (and its vehicle).

Most GPS units sold today are NMEA compatible. That means that not only can they display latitude, longitude, and time on a LCD screen, but the information can be presented to your computer or TNC via the GPS unit's serial data port. The information is presented in ASCII in the form of NMEA "sentences". A number of these sentences are available —with varying kinds of information, but the one used most often for amateur radio activities is "$GPGGA". This sentence contains latitude, longitude, altitude, and time (in UTC, derived from the satellite clocks). The TNC, in GPS Mode, can receive and retransmit this information as beacons and/or store these sentences in a large tracking buffer for later retrieval. The TNC, in GPS Mode, also updates its clock to UTC, derived from the satellites. Hence, the location of your remote packet station, recorded at a specific time, can be tracked by other packet stations.

Amateurs the world over have discovered GPS and are combining this exciting new technology with packet radio for reporting the position of their vehicles, following balloon launches, finding hidden transmitters, and determining location information for more serious applications as well. A typical amateur location reporting system consists of GPS-packet equipped vehicles and fixed-location packet repeaters. At a minimum, each vehicle would carry a GPS receiver, a packet unit (TNC), and a transmitter. Each TNC, in GPS mode, would be set to beacon location periodically. The fixed-location digipeaters would relay the reporting stations' locations over a wide area. For more details on setting up such a system, see the next section, on "Advanced Digipeating."

GPS Equipment (or weather station) Requirements

To set up a packet GPS station that can be tracked, you'll need the following equipment:

- a transceiver and antenna,
- A Kantronics TNC that supports the NMEA 0183 interface standard.
- a GPS unit with an NMEA interface and portable or external antenna. More than 50 vendors produce GPS units and most offer NMEA data interfacing which is essential.

- Over time, a number of software programs will be developed by amateurs to use packet and GPS. One shareware program that has emerged already is the Automatic Packet Reporting System (APRS), written by Bob Bruninga. An APRS system may consist of a GPS unit, a computer, and (optionally) a packet unit. In APRS, NMEA data, from a GPS unit, is passed to the computer, and it is the computer's job to display the information and, if desired, transmit that location via a packet unit using unproto packets.

- In some applications, you may wish to keep your PC or terminal attached to the TNC for packet operations, yet switch serial port connection to the GPS unit periodically—just to update your location. This arrangement is handy when only one COM port is available on your PC. This can be accomplished with an APRS-HSP cable and APRS software.

Cabling a GPS Unit to a Kantronics TNC

The NMEA standard specifies that the signals in the GPS receiver data port follow the EIA-422 standard.

Technical note*: Many GPS vendors simply use "TTL" voltage levels (+5 V and 0 V), but use the same polarity as RS232. These units are capable of driving a Kantronics TNC RS232 serial data port input pin. However, some GPS vendors invert the sense of this signal (true TTL), and these units will not directly interface with a TNC.*

Warning: The Kantronics TNC serial ports are RS232; that is, they provide for signals that swing plus and minus approximately 8 V. Check your GPS manual before connecting this port to your GPS device to make sure that it is capable of handling these voltages. Kantronics is not responsible for damage wherein its TNCs are attached to other devices; see the Kantronics Limited Warranty.

To connect the data port of a GPS unit to your TNC, you'll need a three-wire cable. A shielded cable is preferred. Refer to your GPS manual to determine which pins are used for signal ground, data output to an RS232 device, and data input from an RS232 device (such as the TNC). At least two of these wires must be connected at both ends of the cable:

- connect the signal ground wire of the GPS data port to pin 7 of the TNC serial port. This wire may be identified as "NMEA-" by some GPS vendors.

- connect the “data output pin” of the GPS data port to pin 2 of the TNC serial port. The data output pin may be identified as “NMEA+” by some GPS vendors; this is the line that supplies the NMEA sentences to your TNC.

Some GPS units may allow for programming of parameters. If so, this is accomplished by passing commands to the GPS unit from a terminal (or TNC) via an additional pin in the data port.

If your GPS unit accepts the setting of parameters, you may wish to connect the third wire of your cable from the input data pin of the GPS data port to pin 3 of the TNC serial port. Again, consult your GPS manual for the name of this pin; names vary by vendor. Also, make sure this pin will accept an RS232 signal (+ and – voltages).

Alternate GPS Input

In the KPC–3 Plus, an alternate input for GPS data was added in firmware version 8.3. This input is pin 2 on the KPC–3 Plus’ “Radio” port, and is enabled with the KPC command GPSPORT. A data signal ground is also required, which is pin 6 on the “Radio” port connector, or pin 9 (if internal jumpers are set to configure pin 9 as a ground connection).

Note: The default function for pin 2 on the KPC radio port is XCD (external carrier detect). If pin 2 on the KPC radio port has been connected to a GPS device, and GPSPORT has been configured for GPS input on this pin, the XCD function is not available.

Configuring a Kantronics TNC for GPS Operation

To configure your Kantronics TNC in GPS Mode you must set several command parameters from your PC keyboard as follows:

- select the GPS strings you will be capturing in each of the Location Text buffers
- set the beacon rates for each Location Text buffer you are using
- enter the unproto paths for each Location Text buffer you are using

You determine which GPS strings will be stored in the Location Text (LT) buffers by entering the string header with the GPSHEAD command. The TNC searches for specific strings of data from the GPS unit and stores them in up to four LT buffers. These LT buffers may then be transmitted at periodic intervals as a beacon. The frequency of the beacons is controlled independently for each LT buffer, by the BLT command (which stands for **B**eacon **L**ocation **T**ext). You may also set the destination address and the path for each of the four LT beacons independently using the Location Text Path (LTP) command.

- If the GPSHEAD string for any LT buffer is empty, that LT buffer can be set up as an auxiliary beacon; LT, BLT, and LTP are used like BTEXT, BEACON, and UNPROTO.

In addition to transmitting the beacons periodically, your TNC can store the transmitted beacons in a location buffer (LTRACK) for later review. The size of the LTRACK buffer is set by a command from the keyboard.

Setting Parameters in the TNC

- To add GPS capability to your portable packet station, enter the required GPS parameters into your TNC (from your PC), set ABAUD to 4800, and then cable your TNC to the NMEA data port of your GPS unit.

Note. Most GPS units have a fixed NMEA data port rate of 4800 baud; however, some advanced units allow other bauds.

To set up your TNC for GPS operation, you must first be communicating with it via your terminal in the normal mode (Command Mode or with a Host Mode program).

First, set the GPSHEAD command to capture specific sentences from the GPS unit. You can set up to four strings to be stored (one each) in the LT buffers. The command would be **GPSHEAD n string** where n is the LT buffer number, and string is the name of the GPS string to be stored.

For instance, if you want the TNC to store the $GPGGA string in LT buffer number 1, use the command:

```
GPSHEAD 1 $GPGGA
```

In order to beacon an LT buffer, you must set the LTP and the BLT commands for that LT buffer. Full details of the LTP and BLT commands can be found in the commands section, but here's an example of how they could be used:

If you want LT 1 to beacon with a path of GPS via a digipeater (called DIGI), and you want the beacon to occur every 30 minutes, use the following commands:

```
LTP 1 GPS via DIGI
BLT 1 EVERY 00:30:00
```

- Several amateur groups, such as APRS (see the bibliography at the end of the GPS section), have developed special programs for enhancing amateur GPS activities. Check with them for current practice in setting GPS beacon paths, beacon rates, and frequencies used in your area.

You can also set up a "tracking buffer" to store the most recently transmitted LT information for later retrieval. The size of the buffer and which LT buffers to store is controlled by the LTRACK command. For instance, the command

```
LTRACK 5 LT1 LT2
```

would set the size of the LTRACK buffer to approximately 5 kbytes and store the LT 1 and LT 2 buffers whenever they are transmitted (as set by the BLT command).

GPS Unit Initialization from the TNC

Some GPS receivers require reception of a special string of characters for initialization before they will send any information. You can tell your TNC to send the initialization string to your GPS unit by entering the required string using the GPSINIT command. For instance, if your GPS receiver requires the string $PGRMO,GPGGA,1 to be sent, you would give the command

```
GPSINIT $PGRMO,GPGGA,1
```

to your TNC. Upon power up, the TNC will then send that string via its serial port to the GPS unit (when the TNC INTFACE command is set to GPS.)

GPS Operations

Starting the TNC in GPS Mode

- Hint: Be sure to set ABAUD to that required by your GPS unit.

Once you have the TNC parameters set for GPS, you need to set the TNC to start in GPS Mode on power up. To do this, set the INTFACE command to GPS. When the TNC resets (either from a RESET command or by turning the TNC off and then on) it will be in the GPS Mode and the GPSINIT string (if any) will be sent to the serial port.

Normally, you would set the INTFACE command to GPS, turn the TNC off, connect your GPS unit, and then turn the GPS unit ON and then turn the TNC ON.

When operating in the GPS Mode, the TNC will automatically update the LT buffers every time it receives the designated NMEA strings from the GPS receiver.

Exiting GPS Mode

If you want to exit the GPS Mode (perhaps to set new parameters), connect your computer or terminal to the serial port of the TNC and type three <Ctrl+C> characters.

The TNC will reset and the INTFACE command will be set to TERMINAL.

Hint: You may have changed the ABAUD setting (usually to 4800) for the GPS mode. If so, before you can use the three <Ctrl+C> characters to exit GPS mode, you must set your RS232 baud in your computer to the same as the ABAUD setting in your TNC.

Other GPS Features

Slotted Beacons

The BLT command provides for the ability to specify not only the interval between beacons, but a starting time for the beacons. This permits a “slotted” system of beacons by numerous stations. For instance, lets say your local area has 30 stations that each should beacon their location every 30 minutes. A beacon would be transmitted every n

minutes from the time the user sets the time that the beacon should be transmitted. As an example, if you set the BLT command for LT 1 (BLT 1) with the command:

```
BLT 1 EVERY 00:30:00 START 00:00:01
```

the TNC would transmit a beacon at 00:00:01 (time after midnight) and every 30 minutes after that. The second user could set the BLT with the command:

```
BLT 1 EVERY 00:30:00 START 00:01:01
```

In this case, these two users would never beacon at the same time, in fact they would beacon 1 minute apart. Using this system for our example, it is possible to set all thirty users to beacon one minute apart, avoiding collisions.

Tracking without Beacons

You may configure the TNC to store the LT “beacons” in the LTRACK buffer without transmitting the beacons. To do this, set the LTP to NONE for the associated LT buffer:

```
LTP 1 NONE
```

When the destination call (path) is set to NONE, the TNC will not transmit the beacon, but since the BLT command is still active, the data will be stored in the LTRACK buffer.

Remote Access and GPS

When the TNC is in GPS Mode, the serial port is expecting data only from a GPS unit, hence, it cannot process any “normal” commands. Should you need to change any of the TNC commands, however, there are two options: 1) exit the GPS Mode using a terminal, or 2) connect to the TNC remotely by radio (MYREMOTE) and change the commands.

When you connect remotely to the TNC (and properly respond to the password security check) you can change any parameters including the GPS parameters. You may also wish to send a command string to your GPS unit. This is accomplished with the RPRINT command. For example, to send the string $PGRMO,GPGSA,1 to the GPS unit, connect to the MYREMOTE of the TNC, verify the password, and send

```
RPRINT $PGRMO,GPGSA,1
```

to the MYREMOTE. The TNC simply sends this string to its serial port (which is connected to the GPS receiver).

Other Notes

The LT buffers, LT beacons and LTRACK buffer can be used even without a GPS unit attached. You can manually set the LT buffers to any text you want (up to 128 characters total) using the LT command:

```
LT 1 This is ltext buffer number 1
```

The LT buffers will be transmitted based on the BLT and LTP commands and stored according to the LTRACK command even when the TNC is not in GPS Mode.

GPS Command Summary

The following GPS commands are listed here for convenience and explained in detail in the Command Reference section.

BLT n{EVERY | AFTER} hh:mm:ss [START hh:mm:ss] (n = 1 - 4)

This command sets the interval between beacons for the associated LT string.

GPSHEAD n string (n=1-4) (string up to 8 chars)

This command determines which GPS NMEA sentences will be stored in the LT buffers.

GPSINIT string (string up to 128 characters)

This command establishes a string that will be sent to the attached GPS unit upon power-up (i.e., initial text sent to terminal in GPS Mode).

LT n text (n = 1-4) (text up to 128 chars)

This command fills the specified location Text (LT) buffer with “text”.

LTP n dest [via call1[,call2,...]] (n=1-4)

This command sets the destination callsign and the digipeaters used to transmit the LT strings.

LTRACK n [LT1] [LT2] [LT3] [LT4] [SYSOP] [TIME]

This command allocates memory for a special tracking buffer used to store LT messages.

RPRINT text (text up to 128 characters)

This is an immediate command which sends the “text” string from the serial port of the TNC to the attached device (e.g. GPS unit).

Also, see the GPS setting for the INTFACE command. When this command is set to GPS, the TNC will enter GPS mode upon power up. Data from the serial port will be parsed according to the GPSHEAD command, text will be placed into the LT buffers, and beacons will be transmitted according to the setting of the BLT commands.

- **Hint:** To have the TNC exit GPS Mode, connect a PC or terminal to it and issue three <Ctrl+C> characters.

Advanced GPS (or APRS) Digipeating

The “UI” digipeating commands described in this section add flexibility in reporting GPS (or APRS) station positions. It is an established procedure of the reporting stations to digipeat with generic callsigns and through a number of repeaters (e.g., “APRS” repeaters). In this way, reports of their positions can reach out, via a number of digipeaters, to any wide area. At the same time, the digipeaters supporting these activities must establish their TNC parameters such that all UI frames “copied” are digipeated often enough (enough repeats of each frame), but not too often (too many repeats of each frame), to achieve the desired result.

The commands introduced here (i.e., UIDIGI, UIFLOOD, UITRACE, and UIWAIT) allow each digipeater owner a wide range of ways in which to configure their TNC (in cooperation with other digipeaters in a network, who all need to work together) to establish an optimum GPS (or APRS) repeating network.

- Note: This section assumes you are familiar with GPS, digipeating, and APRS (each topic is discussed elsewhere in this manual). For details about APRS (A Packet Reporting System) developed by Bob Bruninga, WB4APR, consider obtaining a copy of “Getting On Track with APRS,” by Stan Horzepa, 157 page paperback, published in 1996 by the ARRL, 225 Main Street, Newington, CT 06111.

Amateurs are using GPS receivers, a packet unit, and an FM transmitter to report the location of their vehicles or stations to others. Their broadcasts use one-way (unproto UI) packets, reporting call sign, latitude and longitude. Packet stations hearing these broadcasts can monitor the location of the vehicles.

A common goal of these position reporting stations is to digipeat a UI packet over a wide area (or a long distance), while moving though areas where the local callsigns may be unknown. An effective solution for not knowing local callsigns is to establish standard generic callsigns for digipeating stations and then use these names in digipeating paths. Of course, this requires the acceptance and use of standards for naming and using digipeating stations, by both the position reporting stations and the digipeating stations.

- Common standards have been established by some local APRS groups, with names such as RELAY, WIDE, and TRACE. We encourage you to contact your local GPS (or APRS) interest group before configuring your TNC with its new digipeater capabilities. Usage may vary by area.

Improving Efficiencies of Advanced Digipeating

As described above, mobile location reporting stations are using digipeating paths containing generic digipeating names (e.g., RELAY, WIDE, TRACE) to extend their range of reporting. Using generic digipeating enables a mobile station to use the same digipeating path as it moves around, since there will (presumably) always be nearby GPS (or APRS) digipeating stations with one or more of these generic names to pick up transmissions (i.e., UI packets) and re-transmit them to a wider area.

Using generically named digipeaters does result in wider coverage of GPS stations, but several inefficiencies may arise from this common practice (depending upon the availability and names of the digipeaters).

Among the problems that may arise in this situation are that: (1) UI packets can be repeated much more than necessary, by the same or by different digipeating stations with the callsigns RELAY, WIDE, or TRACE clogging up the airwaves, and that (2) several stations may re-transmit the same packet at the same time, resulting in a combined signal that cannot be copied by other stations.

Kantronics' TNCs have a suite of UI digipeating commands that can be used to deal with inefficiencies that can arise when a number of digipeaters are transmitting in the same area, at the same time, using the same generic names.

Overview of UI Digipeating Commands

The following UI digipeating commands support advanced GPS (or APRS) digipeating capabilities.

The UIDIGI command may be used to set up to four additional aliases/call signs for "special" digipeating service. To-be-digipeated packets received containing one of these aliases will be repeated (once) with the call sign (MYCALL) of the digipeater substituted for the alias in the digipeated frame. See the example shown below and the command description for more detail.

UIFLOOD and UITRACE are a bit more exotic. Each provides for multi-hop digipeating with just one digipeater address per packet, thereby keeping the transmission time short. For example, to digipeat through three TNCs supporting the UIFLOOD command, the reporting station might set a GPS position path as follows:

```
LTP 1 GPS via wide3-3.
```

A digipeater TNC supporting "wide" set by the UNIFLOOD command and hearing the reporting station's transmission would then digi the UI location packet (assuming it had not done so already, within a preset time), using an address of wide3-2. In turn each similar digipeater down line would digi the reporting station's UI packet and reduce (decrement) the ssid of the digipeater address again. A TNC using UIFLOOD has the option of inserting MYCALL, creating two rather than one digipeater addresses in each transmitted digi packet.

With UITRACE, each time a packet is digipeated, each TNC adds its MYCALL, thus creating a "trace" or return path. In effect, the size of the packet grows by one digipeated address with each hop. A parameter, n, sets a delay time "do not digipeat within n seconds." Again, see the examples below and the command descriptions for detail.

The current setting of UIDWAIT determines whether or not a delay is added to UI digipeat packets (those formed by UIDIGI, UIFLOOD, or UITRACE) before transmission, once the channel is clear. If UIDWAIT is ON, the delay is determined by slottime or persist settings.

The purpose of the UIGATE (which is in multi-port devices only) is to prevent heavy high speed UI frame activity from congesting (“flooding”) the low speed (port 1) frequency.

Each of these commands is documented in the “Command Reference”.

Note: Digipeater priority for call signs is as follows: UIDIGI, UIFLOOD, UITRACE, MYCALL, MYNODE, MYALIAS (e.g. if you assign the same call sign to UIDIGI and MYALIAS, a to-be-digipeated frame with that call sign will be digipeated according to the rules that apply to UIDIGI).

Using “UI” Digipeat Commands: UIDIGI, UIFLOOD, and UITRACE

The following are actual monitored outputs of a lab system consisting of one position reporting station and three digipeaters, where each digi is configured using the UIDIGI, UIFLOOD, or UITRACE command and where UIDWAIT is set ON. We set UIDWAIT ON to force the system to digipeat all or most of the UI frames without collision. In all examples, the reporting station, WØXI, is used to launch a UI packet with the path set by the UNPROTO command.

With UIDWAIT OFF, several digipeaters would transmit at the same time, resulting in corrupted packets. In actual on-the-air use, a system of digipeaters may work well with UIDWAIT OFF, depending upon their mix of location and transmitter power.

- In actual use, GPS (or APRS) reporting stations would set their path with the LTP

Configuring Three Digis Using the UIDIGI Command

Here, three digis, with MYCALLs of A, B, and C, are configured with aliases of RELAY, WIDE, and TRACE (using UIDIGI); and UIDWAIT is set ON. For example station A’s UIDIGI aliases are set as follows:

```
cmd: UIDIGI ON RELAY, WIDE, TRACE
```

The reporting station path is then set to GPS via RELAY, WIDE, TRACE, and a UI packet is launched.

Any one of the stations monitoring will then display the resulting action as follows:

```
cmd: WØXI>GPS,RELAY,WIDE,TRACE: <UI>:5
WØXI>GPS,B*,WIDE,TRACE: <UI>:5
WØXI>GPS,A*,WIDE,TRACE: <UI>:5
WØXI>GPS,B,A*,TRACE: <UI>:5
WØXI>GPS,A,B*,TRACE: <UI>:5
WØXI>GPS,C*,WIDE,TRACE: <UI>:5
WØXI>GPS,B,C*,TRACE: <UI>:5
WØXI>GPS,A,C*,TRACE: <UI>:5
WØXI>GPS,B,A,C*: <UI>:5
WØXI>GPS,A,B,C*: <UI>:5
WØXI>GPS,C,A*,TRACE: <UI>:5
WØXI>GPS,B,C,A*: <UI>:5
```

WØXI>GPS,C,B*,TRACE: <UI>:5
WØXI>GPS,A,C,B*: <UI>:5
WØXI>GPS,C,A,B*: <UI>:5
WØXI>GPS,C,B,A*: <UI>:5

All digis hear the first UI frame and store it. Due to the setting of UIDWAIT, some stations may wait longer than others to digipeat; hence, when they do, they may have several frames stored up - for example, the original and a digipeat from another station. Trace station B as an example. It digipeats the original UI frame from the “reporting station,” swapping its MYCALL for RELAY (B) and marking it (*), then repeats a frame from A and one from C, and, finally, repeats two more (from A,C and C,A) —a total of five! Note that each repeater digis five times so the total number of digipeated packets is 15!

Configuring Three Digis Using the UIFLOOD Command

Here, three digis, with MYCALLs of A, B, and C, are configured with an alias of WIDE, and UIDWAIT is set ON. For example station A’s UIFLOOD call is set as follows:

cmd: UIFLOOD wide,30,ID

The reporting station path is then set to GPS via WIDE4-4, and a UI packet is launched. Any one of the stations monitoring will then display the resulting action as follows:

WØXI>GPS,WIDE4-4: <UI>:5
WØXI>GPS,A*,WIDE4-3: <UI>:5
WØXI>GPS,B*,WIDE4-3: <UI>:5
WØXI>GPS,C*,WIDE4-3: <UI>:5

Note that all three digis see the UI frame addressed to GPS via WIDE4-4. They then, in turn digipeat that frame, inserting and marking as digipeated (*) their MYCALL, and include the new to-be-digipeated and decremented field of WIDE4-3. Note that each digi hears the UI frames repeated by the others but does not digipeat those in turn since a timeout of 30 seconds was specified by the UIFLOOD command. See the command specification for details.

Configuring Digis Using the UITRACE Command

Here, digipeaters A, B, C, and D are set with UITRACE trace, 30. Digipeater B then launches a UI frame using UN GPS via TRACE4-4. The following results:

B>GPS,TRACE4-4/1<UI>:QWER
B>GPS,A*,TRACE4-3/1:<UI>:QWER
B>GPS,D*,TRACE4-3/1:<UI>:QWER
B>GPS,C*,TRACE4-3/1:<UI>:QWER

Stations A, D, and C hear B and digipeat the UI frame, inserting their MYCALL and decrementing the SSID of trace. A does not re-digipeat D or C because its “n” timer is set to 30 seconds, that is, it will not repeat any frame again within that time that includes the same source or destination and data (in this case QWER).

Configuring Digis for HF/VHF Gateway Operations

- This example assumes the use of at least one multi-port TNC that supports both HF and VHF operation.

In this example, we demonstrate cross-band (gateway) digipeating, using selected Kantronics' TNCs. A UI frame is launched on HF and repeated on VHF four times. A KAM Plus and two KPC-3 Pluses (with call signs C, B, and D) are configured as digipeaters and another KAM Plus (call sign A) is set to launch a UI frame from its HF port. The digis are configured with an alias of FLOOD and UIDWAIT is set ON. In addition, MYGATE for digi C (the KAM Plus) is set to C-3 and its UIGATE is set ON/OFF; that is, it is enabled to digi from HF to VHF but not from VHF to HF. The KAM Plus, with call sign A, is then used to initiate an unproto packet with the path GPS via C-3, FLOOD4-4. Either KAM Plus will then display the resulting action as follows:

```
A>GPS,C-3,FLOOD4-4/H: <UI>:234
A>GPS,C-3*,FLOOD4-4/V: <UI>:234
A>GPS,D*,FLOOD4-3/V: <UI>:234
A>GPS,C*,FLOOD4-2/V: <UI>:234
A>GPS,B*,FLOOD4-3/V: <UI>:234
```

Station C hears the HF unproto packet of station A and digipeats it via its GATEWAY out its VHF port. Digis B and D hear the VHF UI packet and store it for digipeating, and D transmits first. In turn station C hears station D's repeat of the frame and digipeats the frame, decrementing flood to "FLOOD4-2." It didn't digipeat its own "FLOOD4-4" frame because it can't hear itself! Station B finishes last due to the slot/persist algorithm. D digipeats the frame only once even though it hears both B's and C's transmissions since they are within 30 seconds of its own transmission.

Bibliography

For more information on the Global Positioning System, we recommend the following resources:

General Information

Bruninga, Bob, WB4APR, "Automatic packet reporting system (APRS)", *13th ARRL Digital Communications Conference Proceedings*, 1994. (address: ARRL, 225 Main Street, Newington, CT, 06111).

Horzepa, Stan, "Getting On Track with APRS," ARRL (225 Main Street, Newington, CT 06111) 1996.

Sproul, Keith, WU2Z, "MacAPRS", *13th ARRL Digital Communications Conference Proceedings*, 1994.

Technical Information

Collins, J., et al., *GPS Theory and Practice*, Second Edition, Springer-Verlag, 1993.

Etherington, Michael, “FM subcarrier network extends differential GPS nationwide”, *Mobile Radio Technology*, February 1994.

Kaplan, Elliot, “The global positioning system (GPS)”, *Communications Quarterly*, Summer 1994.

Langley, R., “The mathematics of GPS”, *GPS World Magazine*, July/August 1991.

“NMEA 0183 ASCII Interface Standard”, (NMEA 0183 version 2.0), NMEA, P.O. Box 50040, Mobile, AL 36605.

Vendors Supplying GPS Units

Some of the more than 50 vendors currently supplying GPS units include:

- Garmin, 151st St, Olathe, KS 66062
- ICOM America, Inc., 2380 116th Ave. N.E., Bellevue, WA 98004.
- Magellan Systems Corp., 960 Overland CT., San Dimas, CA 91773.
- Sony Electronics Inc., 1 Sony Dr., Park Ridge, NJ 07656.
- Trimble - Mobile Computing Products, 645 N. Mary Ave., P.O. Box 3642, Sunnyvale, CA 94088-3642.

KA-Node

Overview

Most Kantronics TNCs (e.g., KPC-3 Plus, KPC-9612 Plus) include, as a part of their firmware, the Kantronics KA-Node, a packet networking node. If you turn this node on, others may use your station (unattended) not only as a digipeater, but as a node, enabling them to find pathways to other stations and making those pathways more efficient.

KA-Nodes, like other networking nodes such as NET/ROM, operate more efficiently than do digipeaters as a link between two stations. End-to-end acknowledgement of received packets is not required with the nodes; instead they handle errors between each other, rather than from end to end (which can cause extra traffic when errors or interference occurs). A KA-Node, however, is “silent” in that it does not automatically connect to and exchange routing data with adjacent nodes, as do fully-featured nodes such as NET/ROM, X1J, or Kantronics’ optional K-Net. Consequently, users cannot issue a connect to a distant station without knowing the path - as they sometimes can with a full-featured node. At the same time, KA-Nodes are useful in that they are more efficient than digipeaters in using channel time, while not requiring the effort and time required to maintain a full node.

When packet got started in the early 1980s, the initial packet units - terminal node controllers (TNCs) - were designed and coded not only to accept station-to-station connects, but to act as digipeaters for other stations. It was the first attempt at packet radio networking, linking two stations together via several others. It soon became evident, however, that digipeating, particularly through busy channels, was an inefficient method of linking two stations via others. First of all, the station initiating the “connect” had no way to know ahead of time whether or not the digipeating stations were available. Second, the AX.25 protocol called for the station being connected to - at the end of several digipeaters - to acknowledge each packet of the initiating station. Packets not acknowledged (due to collisions) had to be retransmitted by the initiating station AND ALL DIGIPEATERS in the communications chain. As a result, communication often ground to a halt when channels were busy.

To solve these problems, NET/ROM, a PROM-based networking program that was installed inside some TNC models, and a number of derivatives of NET/ROM (e.g., G8BPQ, X1J, KA-Node, K-Net) were developed. These networking programs provide the user connecting to a station via one of these nodes with a choice of pathways to other packet stations and with lists of stations heard.

NET/ROM and several of its derivatives provide for automatic routing of your connect request - much like the phone system today routes your long distance calls. Others, such as the KA-Node, assist you in building a pathway by allowing you to connect to each node in turn through the pathway. All of the nodes correct the “end-to-end” acknowledgment problem mentioned above. This is accomplished by building the pathway with a number of individual “local” connecting links; that is, each link in the

pathway is considered a connect and all acknowledgments of packets are kept local. Hence, if only one link of a pathway (chain) is troublesome, the local nodes take care of it.

The following material is provided to assist you in configuring your KA-Node and telling others how to connect to and use it. A listing of the node commands for users follows.

Configuring Your KA-Node

The following commands are used to configure your KA-Node: NUMNODES, MYNODE, NDWILD, and KNTIMER.

In order to set your TNC for use as a KA-Node several conditions must be met. First, you must allocate the number of circuits (channels) you wish to allow through the KA-Node (see NUMNODES). Each circuit consists of an “IN” and an “OUT” side. Secondly, the callsign assigned to the KA-Node (MYNODE) must be different from the callsign used for you (MYCALL), your alias (MYALIAS), the PBBS (MYPBBS), and remote access (MYREMOTE). By default, your TNC has set MYNODE to your callsign with an SSID of -7.

- To cause your KA-Node to accept a connect request to any SSID of the MYNODE callsign (except those mentioned above), set the NDWILD command ON.

Packets passing through your KA-Node are monitored unless your MYNODE callsign is included in the SUPLIST, if turned ON. If it becomes necessary to disconnect a station from your KA-Node, you can issue the command DISCONNECT MYNODE x (x = channel) from the local terminal connected to your TNC.

If no data passes through an established (connected) KA-Node circuit for n minutes, where n is set by “KNTIMER”, then that circuit will be disconnected. The default time is 15 minutes but you can disable this by setting n to 0.

Each KA-Node circuit allowed will require approximately 4.3 kbytes of memory. If you attempt to set NUMNODES to a value requiring more memory than available, you will receive a message indicating that the value is out of range. The total number of circuits, which may be allocated, will also be affected by the amount of memory allocated to the Personal Mailbox.

- Some Kantronics TNCs, such as the 9612 Plus, are multi-port. These multi-port KA-Nodes can support cross-connects, Xconnects; that is, users may connect on one port and then connect out another port, which may be attached to another radio set to a different frequency.

Using a KA-Node

- The material that follows is meant for those using your KA-Node, i.e. connecting to it. You may use this material as a guide to using someone else’s node or in connecting to your own node from another location (car, another station, via a digi, etc.).

To use the KA-Node as a means of connecting to some other node or end-user, you must first connect to the KA-Node. At the cmd: prompt on your TNC, issue a connect request to the callsign of the KA-Node, let's say it's been named LAW. When you make connection you will see the following messages on your display:

```
*** CONNECTED TO LAW
### CONNECTED TO WILD NODE LAW (NØGRG) CHANNEL A
ENTER COMMAND B,C,J,N,X, or Help ?
```

The *** CONNECTED message is sent by your local TNC to your terminal, and the ### CONNECTED TO NODE message comes from the distant KA-Node. NØGRG is the MYCALL of the station containing the KA-Node in this example, WILD indicates that he is running a "wildcard" node, and CHANNEL A indicates that you have connected to its channel A. If A is in use, you may obtain channel B. The channels, or circuits, are assigned by the KA-Node as needed.

After connecting to the KA-Node, you are in CONVERS mode at your own station, and the KA-Node is waiting for a command. You issue a command to the node by STAYING IN CONVERS MODE.

Note: The KA-Node will interpret the data you send as its commands.

At this point, let's assume that you wish to know what other KA-Nodes are nearby. You would issue the NODES command by typing N, or NODES, in response to the KA-Node "enter command" prompt. You will receive a list of KA-Nodes which have recently been heard. For example, let's suppose that KC was heard by LAW. Your list received from the Nodes command would be:

```
KC (NØAPJ-2) 12/23/87 02:38:45
ENTER COMMAND B,C,J,N,X, or Help ?
```

KC denotes the KA-Node callsign (MYNODE), the MYCALL of the KA-Node station is in parentheses, followed by date and time heard. If LAW had heard nothing, it would respond with:

```
NO KNOWN NODES
```

You may, instead, wish to know what other stations the KA-Node has heard lately. This would be accomplished by sending a J for the JHEARD command. The node will respond by listing its own MHEARD log. The list will contain end user and node callsigns and is the same type of list you get by using your own MHEARD command at the cmd: prompt.

At this point, let's suppose that you would like to connect to the node called KC through your current connection with LAW. Just issue a connect request to KC as follows in response to the "enter command" from LAW:

```
CONNECT KC
```

The response will be:

```
###LINK MADE
###CONNECTED TO NODE KC (NØAPJ-2) CHANNEL A
ENTER COMMAND B,C,J,N,X, or Help ?
```

At this point you are “patched” through the KA-Node LAW to the node KC. When LAW issued the connect request to KC it used your own call but subtracted a count of one from your SSID. For example, if you connected to LAW with WØXI, LAW connected (via your request) to KC with WØXI-15. This is automatic.

At this point, you could connect to another KA-Node, some other network node using AX.25 as an uplink or downlink protocol, or to an end-user. Let’s assume that you desire to connect to NØKN. So, just enter in response to the node “enter command” above:

```
C NØKN STAY
```

and you’ll get the response:

```
###LINK MADE
```

We’ll explain the “STAY” later.

You are now connected to NØKN. If you have issued a connect command to a KA-Node, and realize that you have made a mistake with the call, or for any other reason wish to cancel the connection, you may do so by using the ABORT command. In order for the abort to work, it MUST be the first word entered following the connect command and must be spelled out entirely. For instance if you tell the KA-Node:

```
C KB5EEG
```

and then decide to abort the connect, your next entry must be:

```
ABORT
```

Any other entry will cancel any possible abort of the connection, and you must wait for the KA-Node to retry out (e.g., because it will try to connect to KB5EEG).

Now that you are connected to NØKN, you can carry on a normal packet QSO. While everything appears “normal” and AX.25 compatible, acknowledgments to your packets are generated by the KA-Node directly connected to you. Each link in the chain/pathway takes care of its own errors. In other words, the link between KC and LAW handles its own error checking. In this way, one weak link will not cause end-to-end packets and acknowledgments to be repeated as they would with digipeating. The result is substantial improvement in throughput for connections using nodes.

When it comes time to disconnect, you do so in the standard AX.25 manner. To disconnect the link described above, type <Ctrl+C>, obtain the cmd: prompt on your TNC, and issue the disconnect command:

```
cmd: D
***DISCONNECTED
```

You'll get the usual disconnect message from your TNC, as noted. If however, your distant partner, in this case NØKN, disconnects the link, you'll see the following (due to the "STAY" in "C NØAPJ STAY" in the example earlier):

```
###DISCONNECTED BY NØAPJ AT NODE KC
ENTER COMMAND B,C,J,N,X, or Help ?
```

Frequently when connecting to a distant city, you may wish to talk to more than one station. Perhaps when you connected, you got a message saying "I'm not here right now, please leave a message in my PBBS". By using the "S(tay)" option when telling a KA-Node to connect to another station, the KA-Node will not disconnect from you if it receives a disconnect from the distant station. Instead, you will receive a message from the KA-Node saying ###DISCONNECTED BY (call) AT NODE (MYNODE). In other words, if you connected from Lawrence, Kansas to a KA-Node in Lincoln, Nebraska, and then told that KA-Node to connect to someone using the command "C call Stay". Then if that station sent you the above message and disconnected, you would remain connected to the KA-Node in Lincoln! This would allow you to issue a connect directly to his BBS, without having to re-establish the entire path.

In multiple KA-Node paths, each time you say B(ye) to the distant KA-Node, this will return you to the next previous KA-Node which had been told to "Stay" in the chain of KA-Nodes. From that point, you could build a path in a different direction. One point to note here is that if you use the KA-Node to connect to a BBS (WØRLI or WA7MBL for instance), and use the STAY option, then say B(ye) to the BBS, you would remain connected to the KA-Node closest to the BBS. If you issue the connect without the STAY option, any disconnect from either end will cause the entire link to disconnect.

Automatic Disconnect

If a user abandons a connection to a KA-Node or a link between two KA-Nodes without disconnecting and there is no activity through the link for a specified period of time (see KNTIMER), the node will initiate a disconnect.

Using the XCONNECT Command

Note: The XCONNECT command is available only on Kantronics' multi-port TNCs, such as the KPC-9612 Plus. Single-port TNCs (e.g., KPC-3 Plus) do not support this command, but they may connect to a KA-NODE that does.

The cross-connect (XCONNECT) command is a unique feature of the KA-Node. This command allows cross linking between two frequencies through the node in much the same manner as the Kantronics unique gateway, but with local acknowledgment of packets.

For example, suppose you just connected to node LAWKAN and wish to cross-connect to NØGRG whose station is tuned to the frequency of the other port of the node. Just issue X NØGRG following the node prompt:

```
ENTER COMMAND B,C,J,N,X, or Help
X NØGRG <CR>
```

The response you receive will be:

```
### LINK MADE
### CONNECTED TO NØGRG
```

You can also determine from the response to the NODE command, which port a node is on. Below is an illustration of a KA-Node response to a NODE command:

```
LAWKAN* 12/02/87 15:45:00
N66046/X 12/02/87 15:49:15
OLAKAN/X 12/02/87 16:15:21
```

In this typical display, the callsign of the node is given, followed by the date and time it was last heard. The “/X” indicates that the node was heard on the opposite port from the one you are connected on. The asterisk (*) means that the node was heard via a digipeater.

Determining Which Port You Have Connected To

When you are connected to a multi-port node, you can determine which port you are on, at the node, by using the JHEARD command. A typical node response to the JHEARD command may appear on your display as shown (the KAM uses /H and /V instead of /1 and /2):

```
N66046/2* 12/01/87 14:32:69
NØKN-1/1 12/01/87 16:25:01
WØXI/2 12/02/87 16:28:05
NØGRG/1 12/02/87 16:32:09
```

In this illustration the number following the slash (/) indicates the port the station was heard on. Your callsign will usually be the last one in this list.

You can now see that by comparing the port on which your call appears in the JHEARD list to the port indicated for other stations, you can determine whether to CONNECT or XCONNECT to the station of your choice.

KA-Node Commands for Remote Use

When accessed by radio, the KA-Node supports several commands, which may be given to it. These commands are listed here for reference, with a description of each command. In these descriptions, the UPPER case characters of the command are required, while LOWER case characters are optional. Those items listed within [ ] are optional and if used the UPPER/LOWER case convention applies.

ABORT

Aborts a KA-Node Connect or Xconnect request if it is the first data sent after the connect request. It must be spelled out entirely.

Bye

Causes the KA-Node to initiate a disconnect.

Connect callsign [Stay]

Causes the node to issue a connect request to “callsign” in the usual AX.25 mode. If the connect is successful, a link will be made to the next node or end-user station called. The optional Stay feature provides a way to disconnect without loosing the entire link. Normally when a disconnect occurs, whether issued by you or by the remote station, the connection to the KA-Node is also lost.

Using the Stay option allows you to stay connected to the KA-Node when the remote station issues a disconnect.

Help

Causes the KA-Node to send a brief help list, showing all commands available, with the options and a short description of each command.

Jheard [Short|Long]

Causes the KA-Node to transmit its MHEARD log. The returned list (without the short or long option) may look like this:

```
 (HF/VHF Multi-Port)
LAWKAN/H*    01/09/88 08:25:15
N66046/V     01/10/88 00:03:10
NØKN-3/H     01/10/88 00:03:19
NØGRG/V      01/10/88 00:04:15

(VHF/UHF Multi-Port)
LAWKAN/1*    01/09/88 08:25:15
N66046/2     01/10/88 00:03:10
NØKN-3/1     01/10/88 00:03:19
NØGRG/2      01/10/88 00:04:15

(Single-Port)
LAWKAN       01/09/88 08:25:15
N66046*      01/10/88 00:03:10
NØKN-3       01/10/88 00:03:19
NØGRG        01/10/88 00:04:15
```

The left column indicates the callsign (and SSID if appropriate) of a station heard. The character following the slash (/) shows the port on which the station was heard (single port TNCs don’t include a “/”). The asterisk indicates the station was heard via a digipeater. The center and right columns indicate date and time the station was last heard. The last call on the list will probably be your call. The above JHEARD lists show NØGRG connecting to the KA-Node.

The short (JHEARD S) version of this command would produce the same list, without dates and times.

The long (JHEARD L) version of this command will also show the destination field and any digipeaters used.

Nodes [Short|Long]

Causes the node to return a list of KA-Nodes it has heard, as well as K-Net, G8BPQ, NET/ROM and TheNet nodes. The format of the list is similar to that of the JHEARD list above.

Xconnect callsign

Note: The Xconnect callsign command is available only in KA-nodes in Kantronics' multi-port TNCs.

Causes the node to issue a connect request to "callsign" (in the usual AX.25 format) on the opposite port of the KA-Node. Cross connecting enables you to gain access, via the node, to another frequency.

Introduction to Basic Packet Networking

A network node is a "collection point" in a packet network. It is at these node locations that packets are collected and routed to other nodes and end-users. Many network nodes operate at high data rates, usually at 9,600 or 19,200 baud. End-user access to the network is usually at a lower baud (1200 baud) and on a different frequency. The advantage of a network system is two-fold: automatic routing of connections to other network nodes and increased data throughput. A network node "knows" how to automatically route data through neighboring nodes to distant nodes on the network. They do this by "hearing" other nodes broadcast this information. This means that a user connected to a node does not need to specify the particular route to take in order to reach a distant node as long as the local node contains a listing for that distant node. Using a network node can also dramatically increase the data throughput to distant locations. The packet data and the acknowledgments travel from node *to* node (rather than from end-to-end), greatly reducing the inevitable data collisions that will occur when attempting to communicate over long distances.

The defacto standard networking protocol, NET/ROM, was developed in the late 1980's by Software 2000, Inc. Since then, other node derivatives such as TheNET, TheNET Plus, TheNET X-1J, and G8BPQ have evolved. Although these variants differ in their features, they all utilize the same basic networking protocol.

Some network nodes have more than one radio port in order to provide a high-speed network "backbone" on one frequency, as well as Local Area Network (LAN) access to the high speed network at a lower data rate on a different frequency. This approach to packet networking makes good sense for a number of reasons. First, the high speed backbone is free to pass large amounts of data without competing with end users for the channel. Generators of large amounts of data are called servers. Many times servers are busy communicating with other servers, so dedicating a channel to this function is desirable. Second, users are "shielded" from the server-to-server activity. Many users could peacefully co-exist with a server if all the server did was to respond to user queries. User input to a server generally consists of very small amounts of data that generate fairly large responses. In a local area, this works quite well since all users can detect the presence of the server's data. Carrier Sense Multiple Access (CSMA) allows the channel to be shared nicely. However, servers, and in particular many Bulletin Board Systems (BBSs), spend much more time forwarding and receiving messages from neighboring bulletin board systems than they do actually serving the local user population! It is not unusual for a BBS to attempt forwarding *(and* reverse forwarding) on an *hourly* basis. This situation quickly evolves into quite a paradox - BBSs provide an extremely popular service that has been a boon to the growth and popularity of packet radio. As user numbers increase to take advantage of the services a BBS provides, BBS systems also increase in number in order to provide the service to more and more users. More and more BBS systems must now forward to each other, generating huge amounts of data on the channel, leaving precious little time for the very source of their existence - the user community. User-to-user (keyboarding) QSOs become, at best, a very frustrating proposition. As a result, many users become ex-users, interest declines, and packet usage stagnates. With the K-Net function enabled in your KPC-3, cost-effective networking can benefit packet users in your area immediately.

Basic Networking Guidelines

The intent of this section is not to dictate how to configure a packet network. However, certain goals and concepts of design should be carefully considered. The primary goal of any data network should be the efficient transfer of data from point A to point B (high speed), coupled with ease of use (low frustration level). The efficient transfer of data is made possible with the advent of high-speed modems and "data-ready" radios capable of operating at 9,600 and 19,200 baud. These modems and radios are now available from a number of commercial sources. This means that making internal radio modifications for high-speed data operation is quickly becoming a thing of the past.

User access to the high-speed channel is generally accomplished by using a separate, low-speed port at a network node site on a different channel. This protects users from server-to-server activity and also provides a relatively clutter-free Local Area Network for keyboard-to-keyboard activities.

In an ideal packet network, all high-speed links would operate on a dedicated channel (backbone) without contention. This would require that no stations other than network nodes and servers reside on the backbone channel. Each node on the backbone would have a "rock solid" link to the next, preferably on alternating, or full-duplex channels for maximum data transfer efficiency. All users in a community would be grouped into Local Area Networks (LANs) with each LAN assigned a specific operating frequency such that adjacent LANs would not interfere with each other. Each LAN would have a user port at a local network node that would provide access to servers and other LANs in the region. This situation would provide very efficient server-to-server communication, while offering users an enjoyable environment consisting of a useable keyboard-to-keyboard channel and easy access to servers and distant LANs. In reality, much of the above is quite difficult to achieve. Obstacles include finding suitable node sites, available funds for equipment, and dealing with the differing philosophies and interests of other node sysops. There are many differing opinions on how to go about achieving "Network Utopia", but virtually all of them have one goal in common - make packet radio fun!

K-Net Network node

Quick Start

The factory default values in the K-Net PROM have the network node operation disabled. To enable the K-Net node, the NETALIAS and the NETCALL must be entered. The NETALIAS is usually a mnemonic that gives users an idea about where the node is located (other network nodes do not care what the alias is). For example, SUTNE is a good NETALIAS for Sutton, Nebraska. The NETCALL is the callsign of the node. The NETCALL consists of your callsign plus an optional ssid (Example - KA0DNV-1). It must be different from any other callsign used in the KPC-3 Plus (MYPBBS, MYALIAS, MYNODE, etc.). Use the DISPLAY ID command to make sure that the NETCALL you select is different from the other callsigns that are in use by the KPC-3 Plus. Many times, local "customs" will serve as guidelines for what NETALIAS and NETCALL to use.

Enter the NET ALIAS:

cmd: NETALIAS SUTNE

Set the NETCALL:

cmd: NETCALL KA0DNV-1

This command will cause the KPC-3 to perform a soft reset as memory is allocated for node operation. Your K-Net node is now in operation!

There are six additional "NET" commands that are only available from the command prompt of the KPC-3 Plus. These commands and their current settings can be displayed with the DISP N command.

cmd: DISPLAY N <CR>

NETBUFFS 32

NETCIRCS 5

NETDESTS 25

NETLINKS 10

NETROUTE 5

NETUSERS 5

Other node commands are accessed by connecting to the node, either locally with your terminal, or remotely. If connected remotely, sysop access (see SYSOP command) is required to list and change sysop related parameters. NOTE: RTEXT command in the KPC-3 Plus must be set to allow remote sysop access. Following the above example:

cmd: C SUTNE <CR>

This will cause the KPC-3 Plus to do an "internal connect" to the K-Net node alias SUTNE, much like connecting to your own PBBS. When you connect to your K-Net node in this manner, you will not receive a command prompt (cmd:). Hitting the Enter key will display the commands available to you. To see the current setting of any of the commands just enter the command and hit the Enter key. Whenever you are at this "invisible command prompt" you automatically have access to the entire command set, since you are assumed to be the SYStem OPerator (sysop).

A short help description of each node command is also available by entering H(elp) or ? followed by the node command (Example - To get on-line Help for the ADDNODE command, enter H ADDNODE<CR> or? ADDNODE<CR>. The entire Help contents can be displayed by entering H H<CR>.

Text Messages

Now that the node is operating, you will probably want to enter some text messages that will provide users with some helpful information. These messages are entered by connecting to your node (cmd: C SUTNE) and entering the commands followed by the text. See CTEXT, INFO, and PORTS in the Command Section.

K-Net Node Commands

An asterisk (*) preceding a command indicates a sysop command. All other commands are available to users of the node.

NOTICE - in order to gracefully integrate your K-Net node into an existing network, it is *imperative* that you coordinate your efforts with existing network sysops. There are numerous node parameters that can seriously degrade network performance if set to conflicting values.

The command section contains some networking terms that must be well understood.

NODE - The term node refers to a network node such as a G8BPQ, X1-J, or NET/ROM node. Non-network nodes such as KA-Node, conference bridges, and digipeaters do not apply to this concept of operation.

USER - A user is either a real, live person using a TNC to connect to and use your node, or a server (BBS) that is basically doing the same thing under computer control.

NEIGHBOR NODE - A neighbor node is a node that your node can connect to "direct", without using an intermediate node. Note that digipeater operation is allowed (2 max.). This means that a node can be "out of range", but can still be considered a neighbor since the connect path does not use any other nodes. All known neighbor nodes are displayed with the ROUTES command.

DISTANT NODE - A distant node is a node that is too far away from you to connect to "direct", requiring the use of one or more network nodes to reach. Distant nodes will *not* be displayed with the ROUTES command.

DESTINATION NODE - A destination node is a node that your node knows how to connect to, either directly (i.e. neighbor node), or, in the case of a distant node, by using one or more intermediate network nodes. All known destination nodes (neighbor nodes and distant nodes) are displayed with the NODES command,

All neighbor nodes are destination nodes, but all destination nodes are not necessarily neighbors. Destination nodes will automatically appear in the nodes table by virtue of your node monitoring your neighbor node broadcasts. A neighbor node will automatically be listed in your nodes table and routes table. Neighbor node broadcasts also contain information about other nodes that you may or may not be able to hear "direct". These distant nodes will also be listed in your nodes table if they pass the criteria for being listed (MINQUAL), and if there is sufficient room for them to be listed (NETDEST).

ADDNODE [alias:]call port neighbor [via digi1[,digi2]] quality [obscnt]

This command is used to add a new destination node or to modify an existing destination node in your nodes table. If the optional obscnt (obsolescence count) is set to 0, the node will be *permanently* listed in the nodes table. Such a route will not be updated or dropped from the nodes table by the K-Net auto-routing logic. If the neighbor node callsign entered is not in the neighbor routes table, it is added with the default quality for the specified port. As long as MINQUAL is equal to or less than QUALITY, destination nodes (neighbor nodes as well as distant nodes) will be added automatically to your nodes table as neighbor node broadcasts are received.

EXAMPLE - Addnode KSLVN:WD0EMR-1 1 WK5M-1 200

K-Net responds: Node added with new route

This adds the KSLVN node with the netcall of WD0EMR-1 to the nodes table. KSLVN is available on port 1 (1200 baud) of the KPC-3 using the WK5M-1 (netcall) node as your neighbor. An initial quality of 200 has been assigned to the new node. Unless "locked in" with the ADDROUTE command, this path quality will most likely change since it is recalculated based on node broadcasts that are received. In this case, WK5M-1 was a previously unknown neighbor and the new route was automatically added.

EXAMPLE - Addnode WLAW:W0XI-1 1 W0XI-1 200

K-Net responds: Node added with new route

This would add the WLAW node with the netcall of W0XI-1 to the nodes table. In this case, W0XI-1 is also the neighbor and is available on port 1 with an initially assigned quality of 200.

EXAMPLE - Addnode YORK:WA0CPS-1 1 WK5M-1 70

K-Net responds: Node added

This would add the YORK node with the netcall of WA0CPS-1 to the nodes table. Since WK5M-1 is an existing route (neighbor) a new route is not added.

EXAMPLE - Addnode CRT:KG5DT-1 1 WK5M-1 700

K-Net responds: Node added

In this case, the optional obscnt (obsolescence count) has been specified as zero. This special case is useful in the event that your node is not able to hear this neighbor's node broadcasts (i.e. if a digipeater(s) or gateway is being used). Normally, if a node is not heard from for a period of time, the obsolescence counter will eventually drop it out of the nodes table. However, this node (CRT) will remain in the nodes table permanently, unless removed by the sysop.

ADDROUTE port call [via digi1[,digi2]] quality [!]

This command will add (or modify) the specified netcall of a neighbor node to the routes table. If the optional '!' is specified, the route entry is either locked, if unlocked, or unlocked if locked (toggle). An existing locked route does not have to be unlocked before it can be modified. Locking a neighbor's route is used to assure that an assigned quality will always exist for that neighbor. The locked quality will also be used in the automatic quality computation of other distant nodes contained in this neighbor's node broadcasts to determine if they should also be added to your nodes table. In the event that this locked neighbor disappears from the network, and then returns, it will automatically be assigned the locked quality. During the time that this neighbor is absent, it, and distant nodes associated with this neighbor, may eventually be removed from your nodes table (even if the route is locked) because the obsolescence count for this neighbor could drop to zero. If this neighbor's route is locked, it will still be listed in the routes table, but with a zero count for destination nodes using that route.

EXAMPLE - ADDRoute 1 W0XI-1 110 !

K-Net responds: Route added and locked

This adds the route to the W0XI-1 node to the routes table in the K-Net node. The route is "locked in" and assigned to port 1 with a quality of 110.

Locking a route assigns a fixed *quality* to a neighbor node. This can assure that any connects to a distant node using this neighbor node as a route will take the "hard coded" route, even if the route automatically generated by the K-Net node is different, such as may occur during band openings.

EXAMPLE - ADD Route 1 W0XI-1 110 !

K-Net responds: Route modified and unlocked

Now the W0XI-1 node is "unlocked" since the '!' is a toggle function.

EXAMPLE - ADD Route 1 W4NGA-1 0 !

K-Net responds: Route added and locked

This adds the W4NGA-1 node to the routes table with a locked quality of 0. This node, and any distant nodes associated with it, will not appear in the nodes table since the locked in quality of zero is probably less than MINQUAL.

NOTE - The W4NGA-1 node can be *completely* ignored by putting its NETCALL (W4NGA-1) in the SUPCALLS list and turning LLIST ON (from the command prompt - cmd:). This would prohibit the K-Net from connecting to as well as accepting a connect from the W4NGA-1 node.

BBS [/S]

This command connects a user of the node to the PBBS of the KPC-3 Plus. If the PBBS is not enabled, this command is not operational and will not appear in the node's Help list. If the optional S(tay) parameter is used, the user will be returned to the node after sending the B(ye) command to the PBBS.

BYE

This command causes the node to send a disconnect packet to the user.

CONNECT [call | alias] [/S]

The CONNECT command is used to establish a connection to another node or to another user/server. If connecting to another node that is listed in the nodes table (see NODES command), the KPC-3 Plus will automatically select the path to be used. A connect using the optional /S will use the Stay option - if the remote station issues a disconnect, you will be returned to the node.

If a user wishes to connect to you, and enters just a C (or C /S to Stay), the node will attempt to "connect" internally to the sysop of the KPC-3 Plus. If you have a terminal connected, you can communicate just like a traditional packet session. If you set the CMSG command in the KPC-3 to PBBS, the user will be routed to your PBBS and will be able to leave you a message.

CQ [text] (up to 128 characters)

This command places a user in the CQ mode. The text is transmitted by the node on the radio port if the CQBC command is ON. If another user tries to connect to the callsign contained in the CQ packet during the next 15 minutes, the node will then establish the link back to the user initiating the CQ. Any other text sent by the user that does not begin with CQ will cancel the CQ mode.

EXAMPLE - If a user (WA0CPS) connects to the node and gives the command:

CQ This is Ed in York, Nebraska - Please connect to me for a short QSO.<CR>

The node will transmit a packet addressed *to* CQ *from* the node-modified callsign of WA0CPS-15.

WA0CPS-15>CQ: This is Ed in York, Nebraska - Please connect to me for a short QSO.

The node automatically subtracts Ed's SSID (0) from 15 (15 minus 0 = 15) since the callsign WA0CPS is already being used in the circuit. If Ed's callsign was WA0CPS-2, the node would modify it to WA0CPS-13 (15 minus 2 = 13).

If another user tries to connect to WA0CPS-15, the node recognizes the connect request and the two users are connected through the node.

CQBC ON | OFF

Default ON

This command controls if the node will transmit a CQ broadcast from a user. By default, the CQ broadcast is allowed. For example, if you do not wish any CQ broadcasts to take place, set the command CQBC to OFF.

CTEXT text (up to 128 characters)

Default blank

This command sets a text message that is sent to users when they connect to your node alias (NETALIAS) but not when a user connects to the node callsign (NETCALL). It is also not sent when the user connects to your node by using another network node.

EXAMPLE - CText Welcome to the SUTNE K-Net node, operated by KA0DNV, in Sutton, Nebraska

You may clear this text by entering a % as the ctext (cmd: CTEXT %).

DELNODE [alias:]call port neighbor [via digi1[,digi2]]

This command is used to delete a destination node from your nodes table. If the node you are trying to delete has an alias (showing with the NODES command), you must specify both the alias and the call of the node to be deleted. If this node is the only one using the specified neighbor as a route, and if the neighbor is not locked in the routes table, the neighbor will also be deleted from the routes table.

DELROUTE port call [via digi1[,digi2]] quality

The DELROUTE command is used to delete a neighbor from the routes table. A neighbor route cannot be deleted if there are any nodes in your nodes table using this neighbor as a route.

EXAMPLE - DELRoute 1 W0XI-1 110

K-Net responds: Route deleted

EXAMPLE - DELRoute 1 WK5M-1 110

K-Net responds: Route unlocked

In this case, the route was not deleted, and the "Route unlocked" response simply indicates that the route is not locked in the routes table. This response will be received if this neighbor is used as a route by other destination nodes in your nodes table. In order to delete this route, *all* destination nodes that use this neighbor, as a route must be deleted first.

HELP

This command displays a list of all commands available to the connected user. Giving the HELP command followed by an available node command will display a one-line help message about that command. Entering the HELP HELP command will give a one-line help message of all available commands.

IDINT n **(n = 0 - 255)**

Default 10

The IDINT command sets the time interval (in minutes) between node ID packets *(not* node broadcasts). Setting this command to 0 disables node IDs.

INFO text **(up to 128 characters)**

Default blank

This command sets a text message that is sent to users when they enter the INFO command after connecting to your node.

EXAMPLE - Info Located 3 miles N. W. of Sutton, NE - 50 watts at 100 feet. Hit Enter for available commands or C to Connect to me.

You may clear the info text by entering a ‘%’ as the info text (INFO %).

Intface TERMINAL I BBS I KISS I XKISS I HOST I NET {TERMINAL}

The NET option has been added to the INTFACE command and GPS and NEWUSER have been removed. Setting the INTFACE command to NET and powering the KPC-3 off and then back on allows the serial port to operate as a dedicated "virtual radio circuit". This allows multiple, co-located network nodes to communicate with each other via their serial ports. See the Node Stacking section of this manual for more information.

LINKS

The LINKS command displays a list of current AX.25 links existing in the node. Links displays the callsign of the two connected stations, the AX.25 State (normally 5), the Port number, the Type of link (U = Uplink, D = Downlink, and X = Crosslink), and the AX.25 Version in use (1 or 2).

EXAMPLE - Links

K-Net responds: WK5M KARL S=5 P=1 T=U V=2

This shows that WK5M is connected to KARL, the AX.25 state is 5 (connected), using port 1, it is an *uplink* using AX.25 version 2. Note: An *uplink* is an end user/server connected *to* the node, a *downlink* is *from* the node to an end user/server, and a *crosslink* is from the node to another node.

Possible State Values (S) are:

1 Disconnected

2 Connect in progress

3 FRMR condition detected

4 Disconnect attempt in progress

5 Connected

6 Waiting acknowledgment

7 Device busy

8 Remote device busy

9 Both devices busy

10 Waiting acknowledgment and device busy

11 Waiting acknowledgment and remote device busy

12 Waiting acknowledgment and both devices busy

The next six commands deal with the ISO OSI (Open Systems Interconnection) Reference Model of network structure, in particular level 3 (Network Layer), and level 4 (Transport Layer).

L3TTL n **(n = 0 - 255)**

Default 25

This command sets the limit on the number of times a packet originated by an uplink (normally a user) at this node may be transferred through other nodes. Each time a message is handled by a node, the time-to-live for that message is decremented. If the L3TTL counter reaches zero, the message is discarded. It is possible for network nodes to create circular paths, and therefore a limit is required to prevent a packet from relaying in the node system forever. L3TTL should be standardized network-wide.

L4DELAY n **(n = 1 - 60)**

Default 5

This command sets the delay time in seconds used by the node after receiving a level 4 frame from another node before the ack is returned. Frequently there will be data ready to be passed in the opposite direction shortly after receiving a frame, and this delay permits the acknowledgment to be "piggy-backed" with the data frame, thus reducing the number of transmissions on the channel.

L4LIMIT n **(n = 0 - 65,535)**

Default 900

This command sets a no activity timeout (in seconds) on a node crosslink. When a user connects to another node, the node first establishes a level 2 (AX.25) connect with that node, and then establishes a level 4 circuit on behalf of that user. When the user is finished, the level 4 crosslink is closed, but the level 2 connection remains until the L4LIMIT is reached, at which point it is terminated. When a user disconnects, the level 2 connection between the nodes remains intact and the L4LIMIT timer is restarted. This decreases data congestion if another crosslink request using the same path occurs before the L4LIMIT expires since the node will not have to re-establish the level 2 connect.

L4N2 n **(n = 1 - 127)**

Default 3

This command sets the number of retries used to pass data between nodes (level 4) in the system. After the L4N2 count reaches zero, alternate paths will be tried.

L4T1 n **(n = 5 - 600)**

Default 120

The L4TI timer sets the time in seconds that the originating node will wait before resending a level 4 data packet if no acknowledgment is received from the destination node (end-to-end info-ack). If UTI is set too small, multiple copies of the same packet will exist within the node system even if the first packet was received correctly.

L4WINDOW n **(n = 1 - 127)**

Default 4

The L4WINDOW sets the maximum number of outstanding frames that may exist in the network for a given circuit. When the node has L4WINDOW unacknowledged frames, it will not transmit any more frames until a level 4 info-ack is received. Network nodes may allow up to 127 outstanding frames across the system. The L4WINDOW size is negotiated between nodes when a crosslink is requested, therefore the L4WINDOW size should be standardized network-wide.

MHEARD [S | L]

This command displays a list of stations recently heard by the node. A time stamp is also shown (unless the Short option is specified). If Long is specified, the node displays the *to and from* callsigns of the stations heard, as well as any digipeaters used.

MINQUAL n **(n = 0 - 255)**

Default 70

This command sets the minimum quality required to add a destination node to the nodes table. Making this value higher will decrease the size of the nodes table, making this value lower will increase the number of destination nodes listed. When K-Net hears neighbor node (A) transmit a node broadcast, it computes the quality to distant nodes (B, C, D, etc.) contained in that broadcast by using the quality that is assigned to neighbor node (A). If the resultant computation is less than MINQUAL, the distant nodes (B, C, D, etc.) are not added to the nodes table (see QUALITY command). The default settings of QUALITY = 255/70 and MINQUAL = 70 means that all nodes that are heard "direct" (neighbors) will be added to the nodes table. If MINQUAL is greater than QUALITY, a new neighbor node will not be automatically added to the nodes table, nor will it be displayed with the ROUTES command.

NODES [* | alias I call]

This command displays a list of all known destination nodes. The list displays the ALIAS:CALLSIGN for each destination node. If the optional * is given, hidden nodes (those beginning with #) are also shown. Specifying an alias or call with the NODES command will display up to three possible routes (neighbor nodes) of descending quality that will be used to establish a link to the specified destination.

EXAMPLE - N YORK

K-Net responds:

Routes to YORK:WA0CPS-1 14051 WA0CPS-1 865 1 KA0DNV-1 753 1 W0XI-1

This list shows the three best routes K-Net will attempt to use (in descending order) to connect to the YORK node. The first number is the route quality, the second number is the obsolescence count, and the third number is the assigned port.

NODESINT n **(n = 0 - 255)**

Default 60

This command sets the time interval (in minutes) between nodes broadcasts (not node IDs). A new setting will be in effect after the previous interval has expired. This interval should be set to an agreed-upon value for all nodes in the area. This broadcast provides your neighbor nodes with update information, which they use to add, modify, and delete nodes automatically from their nodes and routes tables. Each time your node performs a nodes broadcast, the obsolescence counter for all destination nodes in your nodes table is decremented by one.

OBSINIT n **(n = 0 - 255)**

Default 5

The OBSINIT command sets the initial obsolescence count for a destination node. When a nodes broadcast is heard from a neighbor, or when a link is established with a destination node, the obsolescence count is set to OBSINIT for that destination node. Each time your node transmits a nodes broadcast, the obsolescence for each destination node is decremented by one. When the obsolescence value drops below OBSMIN for any destination node, that destination node will not be contained in your nodes broadcast. However, it will remain in your nodes table until the obsolescence count reaches zero, at which time it will be removed unless that node had been added (ADDNODE) with an initial obsolescence count of zero.

OBSMIN n **(n = 1 - 255)**

Default 4

This command sets the minimum obsolescence count, which will allow a destination node to be included in your nodes broadcast. The default values of OBSINIT 5 and OBSMIN 4 allow your node 2 chances to receive a "refreshing" nodes broadcast or a connect from a destination node before that destination node is dropped from your nodes broadcast. When the obsolescence count for a node reaches 0, that node is dropped from the nodes table unless it had been added or modified (ADDNODE) previously with an initial obsolescence count of zero.

PORTS text **(up to 128 characters)**

Default blank

This command sets a text message that is sent to users when they enter the PORTS command after connecting to your node.

EXAMPLE - Ports 145.01 MHz @ 1200 baud

This text can be cleared by entering a % as the port text (PORTS %).

QUALITY m/n **(n = 0 - 255)**

Default 255/70

This is a two-port command (Port 0 = RS232 port, Port 1 = 1200-baud radio port) that assigns the quality that is automatically given to a new neighbor node when its nodes broadcast is heard. It is also used to calculate the path quality to distant nodes that are listed in the nodes broadcast from a neighbor node. Port 0 quality defaults to the highest value possible (255) because a back-to-back serial port path is extremely reliable (see Node Stacking section). The quality value is important for two reasons.

The quality of a node (either automatically or manually assigned) will determine how far away a node will be visible in the network. Proper use of the QUALITY parameter is necessary in order to maintain a *useable* node listing by not allowing a node to propagate useless distances over the network.

The QUALITY command can also be used to assure that known "good" routes are chosen by the auto-routing routines. In the event of a band opening, it is possible for distant nodes to be heard "direct" and be automatically assigned a higher quality than normally calculated by using "good" neighbor routes. If a user attempts to connect to this distant node, the first choice route (direct) will probably fail, and then the second best choice (which may actually be the most *reliable* route) will be tried.

The K-Net node will automatically calculate the quality to distant nodes based on the quality that is received from neighbor nodes when neighbor nodes broadcast their known nodes. If your node hears one of your neighbor nodes (A) broadcast a quality of 70 for a distant node (B), your K-Net will automatically determine the quality of the path to distant node (B) that will use your neighbor (A) in the route. This is done by multiplying the quality of the route assigned by you to your neighbor node (A) by the quality of the distant node (B) received within the nodes broadcast from your neighbor (A) and dividing by 256. As an example, if your node has an assigned quality of 70 to your neighbor (A), and receives a nodes broadcast from neighbor (A) containing a quality of 70 to distant node (B), the K-Net will assign a quality of 19 to distant node (B) (70 multiplied by 70 divided by 256 and dropping any fraction of a whole number). If MINQUAL is set to any value greater than 19, distant node (B) will not be added to your nodes table. On the other hand, if the route quality to your neighbor had been modified with a higher value, such as 200, a quality of 54 would then be assigned to distant node (B) (70 multiplied by 200 divided by 256). In this case, if MINQUAL is set to 54, this distant node would be listed in your nodes table. After you determine a particular quality that works best for a neighbor, "locking" the quality (!) will assure that the desired quality will always exits for that neighbor - even if that neighbor disappears from the network and then returns.

ROUTES

The ROUTES command will display the list of *neighbor* nodes. In order for a new neighbor to be contained in the nodes and routes tables, MINQUAL must be equal to or less than QUALITY. The routes table shows the port number, the neighbor callsign, quality value, the number of destination nodes in the nodes table that use this neighbor as a route, if the route is in use, and if the route is locked. For example:

R <CR>

KARL:WK5M-1}Routes:

>1 KA5ZTX-1 110 1

1 WD0EMR-1 63 7

>1 KA0DNV-1 85 15

1 WA0CPS-1 110 12 !

1 W4NGA-1 0 0 !

This Routes listing shows that the KARL:WK5M-1 node can reach neighbor node (KA5ZTX-1) on port 1 with a quality of 110. There is only one destination node (itself - KA5ZTX-1) in Karl's nodes table that can be reached using this neighbor. A right arrow (>) indicates that this route is either in use or has been used within the L4LIMIT time period.

WD0EMR-1 is also a neighbor node for WK5M-1. This node has a port 1 quality of 63. Using the WD0EMR-1 node as a route, Karl's node can reach seven of the destination nodes listed in its nodes table (WD0EMR-1 plus 6 others),

The KA0DNV-1 node has a path that is available to WK5M-1 on port 1 of the K-Net node. The path quality is 85 with 15 destination nodes in Karl's nodes table able to be reached using this neighbor as a route (KA0DNV-1 plus 14 others).

The next neighbor node is WA0CPS-1. It is reachable on port 1 with a quality of 110. There are 12 destination nodes in Karl's nodes table that can be reached using this neighbor as a route (WA0CPS-1 and 11 others). The exclamation point (!) indicates that the route as well as the route quality (110) has been "locked in" by the sysop of the WK5M-1 node.

The last neighbor node listed is W4NGA-1. Since the quality has been set and locked to zero, there are no destination nodes (including W4NGA-1) listed in Karl's nodes table that can be reached using this neighbor as a route. This node will not be listed in Karl's nodes table, and all node broadcasts sourced from this node will be ignored.

STATS

This command displays a summary of statistics concerning level 3 and level 4 activity on this node. A typical display:

KARL:WK5M-1}

Time active 3986 min

Buffers: 200 max, 199 available, 169 min, 0 out

Known nodes: 19

L4 Connects: 26 sent, 5 rcvd

L4 Frames: 205 sent, 392 rcvd, 6 resent, 0 resequenced

L3 Frames Relayed: 2303

This shows that the node has been active for 3986 minutes, there are 200 buffers in the node, 199 of those buffers are available. In addition, it indicates that, at some time, there were only 169 buffers available, and that the node has never run out of buffers. If min Buffers nears zero, or if the buffers have run out, the NETBUFFS parameter should be increased. It further shows that there are 19 nodes listed in the nodes table, there have been 26 level 4 connects sent and 5 received. 205 level 4 frames have been sent, 392 received, 6 resent (retries) and none have been resequenced. The level 3 activity shows that 2303 level 3 frames have been relayed.

SYSOP

The SYSOP command can be used by a remote user to access sysop commands (marked with a single *). When a user gives the SYSOP command, the node responds with three lines of numbers. The user must then convert one of these three lines into the corresponding characters of the RTEXT in order to be validated as sysop. This operates the same as the remote access in your KPC-3 and uses the same RTEXT. (See the "Remote Access" section in your KPC-3 Reference Manual), If connecting to the node with an attached terminal, sysop validation is automatic.

USERS

This command displays a list of users currently connected to the node. It also displays the number of buffers currently available (in parentheses). Uplinks, downlinks, and crosslinks will be displayed by this command.

EXAMPLE -

Node user enters: USERS

K-Net responds:

KSLVN:WD0EMR-1} K-Net Packet Switch v1.0 (145)

Uplink 1 (WA0PWS) <-> Circuit (KSBLC:WZ0M-3 WA0PWS)

Circuit (KSLAW:WK5M-1 WA0CPS) <-> Downlink 1 (WA0CPS-15 KB0VA) Uplink 1 (KA0DNV)

KSLVN:WD0EMR-1} K-Net Packet Switch v1.0 (145)

There are currently 145 buffers available (145) at the KSLVN:WD0EMR-1 node.

Uplink 1 (WA0PWS) <-> Circuit (KSBLC:WZ0M-3 WA0PWS)

WA0PWS is a user that has connected (Uplinked) on port 1 to the KSLVN node. WA0PWS has requested, and KSLVN has established, a Circuit (Crosslink) to the KSBLC:WZ0M-3 node on behalf of WA0PWS.

Circuit (KSLAW:WK5M-1 WA0CPS) <-> Downlink 1 (WA0CPS-15 KB0VA)

A Circuit, (Crosslink) from KSLAW to KSLVN has been established by the request of WA0CPS with WA0CPS-15 (node modified callsign) then Downlinking to KB0VA on port 1.

Uplink 1 (KA0DNV)

KA0DNV has Uplinked to KSLVN on port 1, but has not established a Crosslink or Downlink at this time.

Fine-tuning Your K-Net Node

Although the default parameters of the K-Net node will get you "up and running", there are some guidelines that can enhance the desired operation. Good neighbor relationships are really the key to good networks. This takes the form of proper quality settings for your neighbor nodes, as well as good relationships with your neighbor sysops. If your nodes table has a consistent listing of destination nodes that have reliable paths, users will not be frustrated, and your node will be perceived as working quite well. On the other hand, if your nodes table is frequently allowed to accumulate a large number of destination nodes that are not fairly solid links, it becomes difficult to use, non-responsive, and very unfriendly to the network and especially to a new user.

After the K-Net node has been running for a while using the default parameters, some "tweaking" of commands are in order. *The quality settings for your neighbor nodes will have the largest impact on the usefulness of your node.*

REMEMBER - The QUALITY number determines how far away a node will be visible in the network and the order of automatic route selection.

The routes table contains the information that will help you as you change the default settings. The quality assigned to your neighbors listed in the routes table can be changed to achieve the desired node set up. There are two very general philosophies that will help you determine how to "finalize" and maintain your node.

Static Node

The default values for the K-Net node will cause every neighbor node that you hear to be included in both the routes table and the nodes table (QUALITY = 255/70 and MINQUAL = 70). Initially, this may be a bit of a problem since weak neighbor nodes and their distant nodes may be saved in your nodes table. Try to connect to all the neighbor nodes that are listed in the routes table. Be sure that you have set the number of NETROUTE such that all neighbors will be included. Make note of which nodes respond with a good signal and modify the routes table with the ADDROUTE command to set and lock the quality of these nodes to a number higher than either value automatically assigned by the QUALITY command for either port. For example, if the QUALITY command is set to 255/70, and all good routes are locked at a quality of 115, setting MINQUAL to 115 results in a static node that is only aware of the neighbor nodes that you have "locked in". This node will *never* have any new neighbor nodes *or* distant nodes automatically added to the nodes table. If distant nodes are to be listed in the nodes table, they must be manually added to the nodes table using the ADDNODE command, listing one of the existing neighbors as the route. Since distant nodes may have multiple paths from your node, the preferred route and alternate route(s) to a distant node can be specified with multiple ADDNODE commands. When connections to these distant nodes are requested, your node will *always* use the locked route path(s) in spite of changing band conditions.

Part of the responsibility of being a node sysop is maintenance. It's probably a good idea to set MINQUAL equal to or less than QUALITY from time to time in order to see if any new neighbor nodes appear in your routes table. If a good path to a new neighbor shows up, lock the new node route with your "good" neighbor quality and then reset MINQUAL to its previous value.

Dynamic Node

The default values for QUALITY and MINQUAL will allow *all* neighbor nodes that are heard to be listed in the nodes table. This means that both good and marginal neighbors will be listed in the nodes table. Of course, the nodes table will be larger than with the static node settings, but the benefit obtained is that the node will know when new neighbors appear and then automatically include them in the nodes table. Although distant nodes that may have good routes from your node will not be listed, some sysops feel that this is one method to control the propagation of marginal nodes. Another effect of dynamic node settings is that users can take advantage of band openings that could allow the use of nodes not normally visible on the network. This can lead to some frantic DXing when band conditions permit. The dynamic approach allows the nodes table to be elastic. It will grow as previously unknown neighbors appear, and shrink as the obsolescence counters for each route decrement.

Once the K-Net node has been in operation for a while, and you have noted your "good" and "not-so-good" neighbors, a combination of static and dynamic settings can be used. By locking in "good" neighbors at high quality values, you can force the routing logic to take a preferred route in spite of changing band conditions. As long as MINQUAL is equal to or less than QUALITY, new neighbor nodes and nodes heard by virtue of a band opening will also come and go as the obsolescence counter is either reset or decremented. In all probability, your neighbor nodes will have "good" nodes that your

node should know about, and your neighbor's neighbor will also have "good" nodes, as well as your neighbor's neighbor's neighbor, and so on. Once "good" neighbors have been established, the quality to assign to each neighbor will have to be determined experimentally in order to control the listing of marginal destination nodes that will appear as a result of receiving neighbor node broadcasts.

As a practical point, once a distant node is 3 or 4 RF hops away from your node, having it listed in your nodes table is not a good practice unless the path is extremely reliable. The chances of a user being able to access a node this distant diminishes greatly as the number of RF hops increases beyond this point. Also, allowing destination nodes to appear in your nodes table that are more than 3 or 4 hops away will result in *very* large nodes tables being generated. When your node receives a nodes broadcast from a neighbor node, K-Net will automatically calculate a quality to all the nodes contained in that broadcast. If the computed quality is less than MINQUAL, that distant node is not added to your nodes table. Therefore, it is desirable to have a combination of route quality and MINQUAL that will only allow nodes that are 3 or 4 node hops away to appear in your nodes table.

As an example, if your "good" neighbor node (A) has a "good" neighbor node (B), and node (B) has a "good" neighbor node (C), your node has a good chance of reliably reaching node (C). The computed quality at your node for distant node (C) must be greater than MINQUAL in order for it to be listed in your nodes table. If a quality of 70 is assigned to all neighbors in this example, your node would assign a quality of 5 for distant node (C), computed as follows (it helps to draw this out on paper): The quality from (B) to (C) is 70, when (B) broadcasts (C)'s quality as 70, (A) multiplies 70 by 70 and divides by 256 to arrive at a quality of 19 for (C). When your neighbor (A) does a node broadcast, his broadcasted quality of 19 for (C) is multiplied by your path quality for (A) and divided by 256, resulting in a quality of 5 for distant node (C) (19 times 70 divided by 256). Since 5 is probably below your MINQUAL, distant node (C) will not be listed in your nodes table even though it is probably useable. What to do? You could change the quality to your neighbor (A) to 255. This will result with your node assigning a quality of 18 for distant node (C), but it does not leave much room for a special case route that may be 5 or 6 hops away. This will also cause your neighbor (A) to propagate further than desired in the network; creating useless routing. As a compromise, lock the quality of *(A)* to 140. If your neighbor node (A) would also lock in a quality of 140 for his neighbor (B) and node (B) would do the same for node (C), then your node would now assign a quality of 41 for distant node (C), leaving some "wiggle room" for other nodes as they join the network. Now a MINQUAL setting of 41 will allow these "good" distant nodes to be listed in your nodes table.

If you work through the above exercise a few times using different quality values, it becomes apparent that there are many settings of quality that will produce the desired effect as long as all sysops adhere to a common guideline. However, the quality values used in the above example were not just "picked out of the air": 140 is a *reasonable* quality to assign to a good 1200-baud neighbor, since this allows good distant nodes to appear in your nodes table while also limiting node propagation. It also leaves room in the quality range for ultra-reliable routes.

This is where good neighbor relationships with other sysops in your area become important. Neighbor node quality must be continually examined in order to arrive at a

useable network configuration that makes good use of all possible paths. Coordination of efforts and philosophies should result in an expanding network that is low in frustration and fun to use.

Node Stacking

Although the K-Net firmware in the KPC-3 Plus provides a quick and easy method to increase network usage and efficiency, as the network grows there will be an increasing demand to provide an even higher level of service to the community. This can take several forms: Adding more user ports to the network as existing user ports become heavily loaded, adding dedicated server access ports to "unload" activity from existing user/server shared channels, and the creation of point-to-point dedicated backbone channels to increase overall network performance.

Since knowledge of how to properly configure the K-Net node is assumed, this section will deal only with the physical connections and parameters that are required to create a node stack.

Before deciding to "help" the community, you should be aware of the impact that adding more ports of any nature will have on the existing network. Generally speaking, more available radio ports will provide a more efficient network provided that this effort is well coordinated with the other node sysops in the area that will be affected by your actions. Sysop coordination and cooperation must exist in order for the network to provide the maximum benefit for all users.

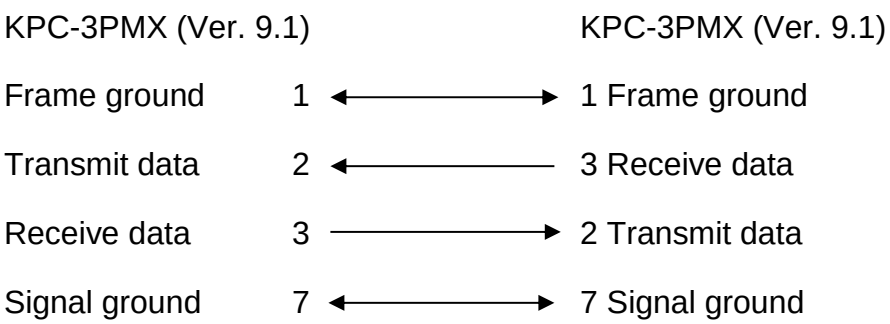
The concept of a node stack is practically self-explanatory. A node stack is formed when two or three or more network nodes are "hard-wired" together (stacked) and allowed to communicate with each other. Since the radio ports are already in use, the serial port of the TNC is used to pass the networking information and data between the nodes in the stack. Setting the INTFACE command (from the cmd: prompt) to NET forces the K-Net node to use the serial port as well as the radio ports for passing network information. Because of this, the serial port can no longer be used as an ASCII terminal port since all communication is now "net-ese" and used by the other nodes in the stack for routing and other network functions. If you need to change a node parameter, the terminal mode can be regained by connecting a terminal to the serial port and using the standard packet transparency escape sequence (three control C method), or the node can be remotely accessed via a radio port if the RTEXT has been set, and the proper response to the SYSOP command is sent to the K-Net node.

Again, the QUALITY parameter plays a major role when node stacking is used. The default QUALITY settings are 255/70 (255 for port 0 - serial port, 70 for port 1 - 1200 baud). The QUALITY setting for the serial port (port 0) is defaulted to the maximum value allowable because the serial link between nodes in a stack is very reliable. Of course this value can be changed, but doing so may cause undesirable node propagation and routing.

The most basic form of a node stack would be to add another 1200-baud port to your KPC-3 node. The additional hardware required is another complete 1200-baud network node station (transceiver, feedline, antenna, TNC to radio cable, TNC with network node firmware such as a KPC-3 with version X.XX N firmware or other TNC with

NET/ROM or TheNET firmware), and a special serial cable to connect the KPC-3 to the other TNC.

Construction of the serial cable is straightforward. Using two male DSUB-25 connectors, connect pin 1 to pin 1, pin 2 to pin 3, pin 3 to pin 2, and pin 7 to pin 7.



With the above cable assembled, the two KPCs with K-Net enabled, are free to exchange network data over their respective serial ports.

Operation is probably very close to what you would imagine it to be. Each TNC must be given a unique NETALIAS and NETCALL because they are separate nodes. They should also be configured with the node parameters that are recommended in your area. The serial cable functions as a "virtual radio circuit" for node-to-node communication in the stack.

REMEMBER! The serial ports are talking to each other - the serial port mode parameter must be set the same in both nodes (i.e. 9600, n, 8, 1).

The final step to activate this 2 port stack is to set the INTFACE command in each of the TNCs to NET and to power them off and then back on.

Three Ports (One High-Speed and two 1200-Baud Ports)

A three port node stack consisting of a high-speed port and two 1200-baud ports can be created by using a KPC-3 Plus modem with version 9.1 PROM installed and a KPC-9612 Plus, also with the version 9.1 installed. Since the high-speed port of the KPC-9612 Plus is capable of 4800, 9600, or 19200 baud operation, this set-up offers tremendous flexibility in network configuration. As above, each of the K-Net nodes must be programmed with a different NETALIAS and NET CALL and use the serial port cable described above to tie them together. Now set the INTFACE command in each TNC to NET and power them off and back on to complete the installation.

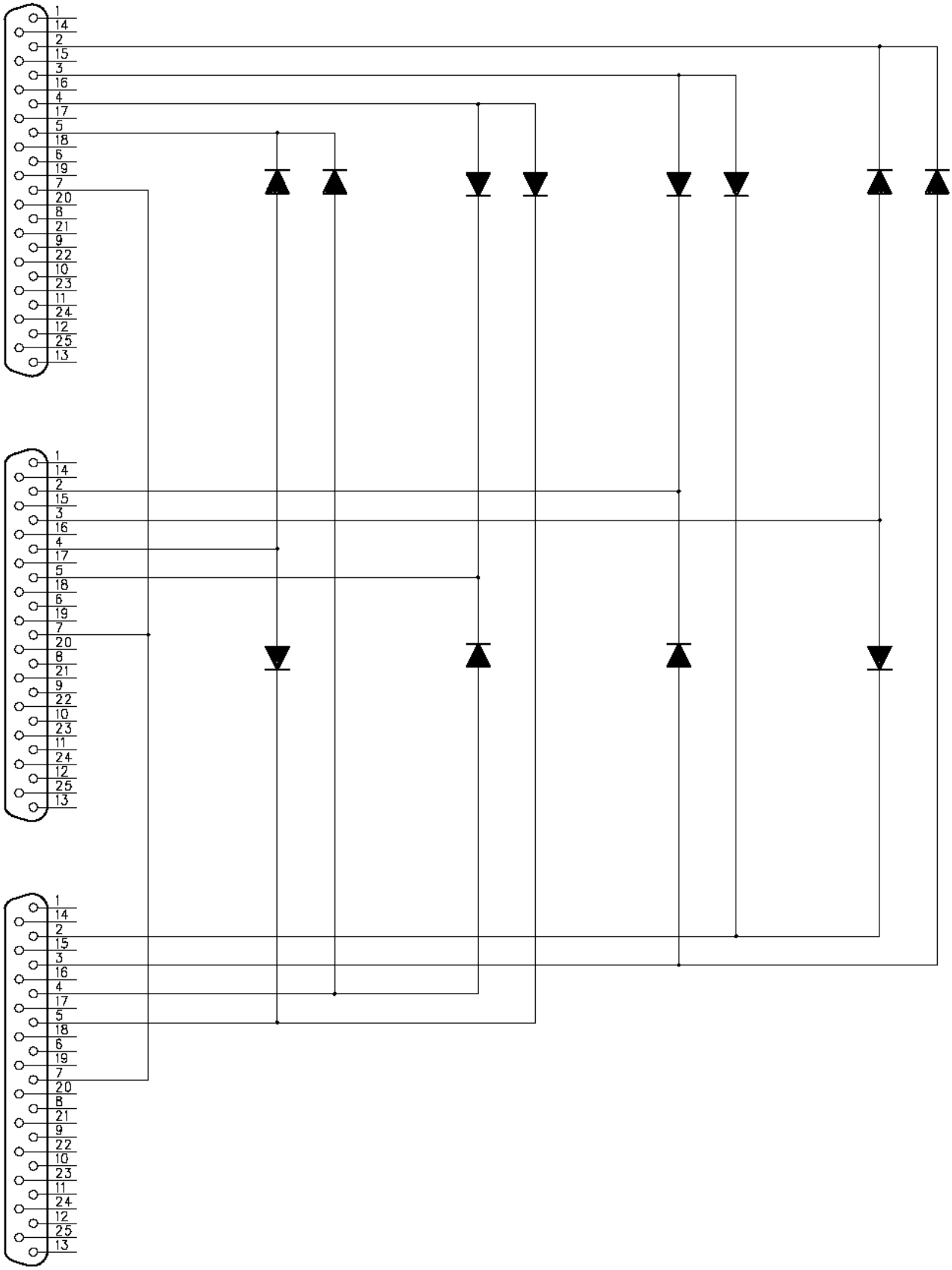
More than Three Ports

The two and three port stacks discussed above are relatively simple to cable together since there is no contention for the serial port. Each node is free to broadcast and pass data over the serial port "radio". Adding another node to the stack means that now there needs to be a method to control access to the common serial port cable in order to avoid data collisions. The K-Net node uses the RTS (Request To Send - pin 4) and CTS (Clear To Send - pin 5) control lines of the serial port for this purpose.

When a K-Net node has data to send to the other nodes in the stack, it first checks the RTS line for a true (logic high) state. If RTS is true, then the node drops CTS and the data is sent to the other nodes in the stack. The dropping of CTS by any node in the stack will inhibit all the other nodes from using the serial port by forcing RTS false (logic low), thus avoiding data collisions within the node stack. The additional hardware required for this operation is commonly referred to as a Diode-Matrix.

K-Net Diode Matrix

A three, four, five, or six serial port Diode Matrix follows the same logic as above, with six ports probably being about the limit of practicality. The number of diodes required for a Diode Matrix with N serial ports is 2*N*(N-1).


1
14
2
15
3
16
4
17
5
18
6
19
7
20
8
21
9
22
10
23
11
24
12
25
13

TheNET X1-J / K-Net Cross-reference Guide

This cross-reference guide is included to assist in the coordination of node parameters if an X1-J node is located in your area. The *basic* X1-J PARMS can be listed by connecting to an X1-J node and giving the P command. This should precipitate meaningful discussion among node sysops in an attempt to arrive at a "Network Standard" of node parameters that benefit all users.

X1-J PARM	K-Net COMMAND
1. Max. Destination Node Size	NETDEST
2. Min. Auto update quality	MINQUAL
3. Neighbor default quality	QUALITY
4. RS232 default quality	QUALITY
5. Initial obsolescence count	OBSINIT
6. Min. Obs. count to broadcast	OBSMIN
7. Node broadcast interval	NODESINT
8. Initial Time-to-Live	L3TTL
9. Transport Frack timeout (sec)	L4T1
10. Transport Retry counter	L4N2
11. Transport Ack Delay (sec)	L4DELAY
12. Transport Busy Delay (sec)	N/A
13. Transport Window Size (frames)	L4WINDOW
14. Transport Overfill Limit (frames)	N/A
15. No Activity Time Out	L4LIMIT
16. Persistence	cmd: PERSIST
17. Slot	cmd: SLOTTIME
18. Link Frack (T1)	cmd: FRACK
19. AX.25 Maxframe	cmd: MAXFRAME
20. AX.25 Retries	cmd: RETRY
21. Link Response Time (T2)	N/A

22. Active Check (T3)	cmd: CHECK
23. AX.25 Digipeat	cmd: DIGIPEAT
24. Validate Callsigns	N/A
25. ID Beacon	cmd: IDINT
26. CQ Broadcasts	CQBC
27. ACL (Access Control List)	cmd: SUPCALLS/LLIST To *completely* ignore uplinks from *and* downlinks to specific user callsigns and neighbor node *callsigns* (not alias) cmd: BUDCALLS/CONLIST To allow uplinks from and downlinks to *only* specific user callsigns and node *callsigns* (not alias)

WEFAX Mode

Overview

Black and white weather facsimile charts referred to as “WEFAX” by HF radio operators, have been transmitted by shore stations for years and received by commercial ships, yachts, and radio amateurs. By coincidence, the signal format used for WEFAX broadcasts, 800 Hz shift key (FSK), is compatible with the modem housed within most Kantronics TNCs (e.g., KPC-3, KPC-3 Plus, KAM Plus). In order to receive WEFAX charts, which are typically hand drawn maps of atmospheric pressure, surface temperature, or wind velocity/direction over a wide range of sea, you must use a special program for your computer. Such programs enable your PC to receive 8 bit data from the TNC and format it properly for display on your computer screen.

These maps, unlike satellite photos with a full gray scale range, are often produced by a particular weather service at their own site. Two predominant stations are NAM, the Norfolk US Navy Fleet Weather Service station, and CFH, the Canadian Forces Halifax station.

Background: HF WEFAX Transmissions and Resolution

Full-size charts are eighteen (18) inches wide and vary in length from 10 to 18 inches. In past years, many charts were generated on a rotating drum at 60 scan lines per minute, but modern day charts are digitized (placed in computer storage) at the rate of 120 scan lines per minute on a flat-bed scanner. The scanner digitizes with a resolution of 96 lines or pixels per inch. So, the maximum resolution of a hand-drawn synoptic chart, when digitized, is nearly 1800 by 1800 pixels!

When the WEFAX maps and charts are transmitted on HF, some of the resolution is lost. This occurs because the FSK modulation scheme used by all manufacturers of HF WEFAX transmission equipment will not support the baud needed for full horizontal resolution. The FSK format has been kept, however, to make today’s equipment compatible with that produced earlier, and received-map resolution is acceptable.

Finding WEFAX Broadcasts

There are approximately 50 commercial and governmental WEFAX transmitters located in over 20 countries around the world. Most broadcast on HF (between 3 and 30 MHz). Many broadcast continually while others are on a sporadic schedule.

These are some typical frequencies where WEFAX signals can be found, if ionospheric conditions permit:

LOCATION	FREQUENCIES (MHz)
Halifax, NS	4.275, 6.630, 9.890, 13.510
Norfolk, VA	8.080, 10.865, 16.410
San Diego, CA	8.646, 17.410
Mobile, AL	9.158
San Fran, CA	4.346, 8.682, 12.730, 17.151

Washington, DC 4.795, 10.185, 12.205, 14.672

In addition, many of the frequency guides for sale by various radio equipment suppliers list WEFAX station frequencies.

Tuning WEFAX Signals on HF

WEFAX broadcasters use an 800 Hz shift key (FSK) format, using mark and space frequencies of 1500 and 2300 Hz respectively. These tones, like voice broadcasts, are used to modulate the station RF carrier. You can tune these signals with your standard 1200-baud packet modem by tuning about 1.7 kHz below the published station frequency in upper-sideband (USB). For example, with your HF transceiver or receiver set to USB, turn the dial until the frequency readout shows 8078.3 to receive NAM (the Norfolk US Navy Fleet Weather Service transmitting on 8080 kHz).

Receiving WEFAX Transmissions

To receive WEFAX you need to connect an audio cable from your HF receiver to the audio input of your TNC's 1200-baud radio port. Be sure to read the documentation that comes with your WEFAX program to learn how to use it.

The WEFAX Command

To cause your TNC to enter the WEFAX Mode, you use the WEFAX command (and NOT the INTFACE Command).

WEFAX n

This command will cause the TNC to sample the audio input at n samples per second. The value of n is determined by the number of pixels (dots) which can be displayed across one line of your terminal screen in the graphics mode. For instance, if the PC computer you use has 640 dots across the screen and the WEFAX station is transmitting 2 lines/s (120/min) you should use an n value of 1280. These values keep your TNC in sync with the HF WEFAX transmitting station. If your WEFAX picture skews (is not squared up), adjust the TNC clock by using the command DAYTWEAK. The ABAUD of the TNC must be set to a value at least 5/4 the WEFAX n rate. This gives the terminal program time to display or store each pixel byte and to handle the extra start and stop bits added to each set of 8-bits at the serial port.

Exiting WEFAX Mode

To stop the sampling of WEFAX data and return to Command Mode, send a <Ctrl+C> to the TNC.

Other Modes of Operation

Other ways of using your TNC are covered in this section.

Some of the following modes of operation require special software, so they may not be available if you are using standard terminal programs such as HyperTerminal or Procomm Plus.

Remote Sensing and Control

You can use two (or more) packet radio stations, each containing a Kantronics TNC, to implement remote sensing and/or remote control, as follows:

- **remote control of another TNC:** from a local Kantronics TNC-based radio station, use the MYREMOTE command to establish control of a remote Kantronics TNC-based radio station and issue commands for the remote TNC to carry out. For details, see the Command Reference.

- **sense analog inputs:** use the ANALOG command to gather and report the status of one or two analog input lines in the remote TNC. Kantronics single port and multi-port TNCs contain an 8 bit A/D converter and additional circuitry that converts the analog voltage presented at two specified inputs into digital values from 0 to 255. You can then use these digital values (reported in binary) as an approximation of the analog voltage values at the time the TNC interrogated the analog inputs. For details on the ANALOG command, see the Command Reference.

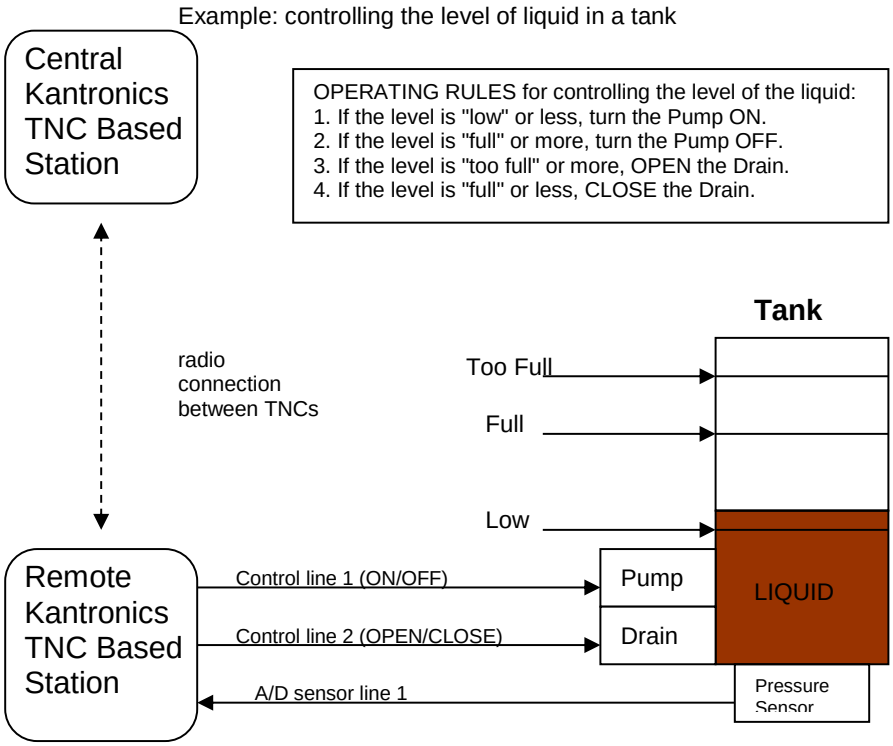
- Note: As documented in the ANALOG command and the radio port pin specifications, single port devices and multiport devices use different input lines and ports for analog sensing.

- **control output voltages:** use the CTRL command to control the voltages on selected output lines in the remote TNC (up to two output lines in single port devices and four output lines in multi-port devices). Each of these output lines can be set to OPEN (i.e., OFF) or GROUNDED (i.e., ON) by the CTRL command. Also, the CTRL command can be used with any or all of these output lines to send a specified number of short ON/OFF pulses (each pulse lasting about 100 ms, with a fixed length pause between pulses) or a single long (about 1.5 s) ON/OFF pulse. For details on the CTRL command, see the Command Reference.

- You can also use the ANALOG and/or CTRL commands, without the REMOTE command, to carry out sensing and control functions at a local Kantronics TNC.

The following diagram illustrates the use of two Kantronics TNC/radio stations for remote control and sensing. These operations could be carried out manually or via a terminal program running in the computer in the central TNC station.

Remote Sensing and Control



As noted earlier, the Command Reference section and the ADVANCED INSTALLATION section provide details needed to adapt these broad capacities to your own situation, with your own sensors, controls, and remote locations.

Modem Mode

In MODEM mode, the TNC demodulates the audio signal from the radio on port 1 and passes this data through to its RS232 port unmodified. No “protocol” adjustments are made to the data; the TNC simply acts as a dumb modem. In other words, it simply reflects out the RS232 port what it “sees” coming in the radio port. For example, if ASCII information is received at 1200 baud, the TNC presents your computer with ASCII at 1200 baud. With a simple terminal program running at 1200 baud too, you’d be able to copy the text.

To place the TNC in modem mode use the INTface command. At the cmd: prompt, type INT MODEM and return. Then turn your TNC power OFF and back ON.

To leave modem mode, send three Ctrl+C (command) characters to the TNC at 1200 baud. If ABAUD was set to something other than 1200, that setting will remain after you exit modem mode. Hence it may be convenient for you to set the TNC to 1200 baud before entering modem mode.

- Note: Some computers may lose some of the data stream if a full RS232 cable is used. This can occur because the RS232 CD line, which is “tied” to the TNC’s receive LED, may be cycling up and down. In this case, set CD to EXTERNAL before entering INT MODEM.

Copying Weather Broadcasts NWS EMWIN

MODEM mode can be used to copy emergency weather broadcasts by the National Weather Service. The NWS Emergency Managers Weather Information Network (EMWIN) is a wireless direct-to-computer information service that the National Weather Service (NWS) provides with help from the Federal Emergency Management Agency. The NWS feeds a constant stream of weather information (@ 1200,8,N,1), from around the world, to a satellite orbiting the Earth. Many cities are coming on line now to capture this stream and rebroadcast it on fixed VHF commercial FM frequencies just above the 2-meter ham band.

Using your TNC in MODEM mode, with a PC and communications software, you can copy these broadcasts. The content of the broadcasts may be plain ASCII text (copyable with HyperTerminal) or graphics. For updated information on EMWIN and software to copy its modes, search the Internet.

To copy the broadcasts, simply cable together the equipment mentioned above, tune your VHF FM receiver to the local EMWIN frequency, set the KPC interface command to MODEM, and follow the text on your PC screen.

Kantronics Host Mode Operation

Kantronics' Host Mode allows your TNC to communicate with computer software that is more sophisticated than standard terminal programs such as HyperTerminal. Software, such as Kantronics' HostMaster II+, that supports this mode will automatically set your TNC INTFACE command to HOST mode and perform a soft reset.

Your host program will provide you with some means to give commands to the TNC. For a complete description of your host program, refer to the manual that comes with your software.

After you switch to HOST mode, your TNC will remain in HOST mode until you switch to COMMAND mode. The easiest way to switch out of HOST mode is to use your host mode software (see instructions with that software). If you begin using HyperTerminal while the TNC is in HOST mode, you will see the message " SØØ ", sent by the TNC to indicate that it is in HOST Mode. If you see this message, you must first exit the Host Mode before you can start talking to your unit with a program such as HyperTerminal. You must send three characters to the TNC in order to exit the Host Mode and go back to COMMAND mode. First, send a FEND character (ASCII code 192), then the letter q (upper or lower case is ok), and finally another FEND character. Your TNC will then leave Host Mode, switch to COMMAND mode, and send the usual Kantronics sign-on message. If you use a terminal program other than HyperTerminal, it may send these characters automatically, or it may require that you carry out this same procedure to get out of HOST mode and to COMMAND mode. See Appendix D for details on this procedure.

KISS Mode

Amateur packet radio communications can use the TCP/IP suite of protocols instead of AX.25 by switching to KISS mode and running software that supports the KISS code designed by Phil Karn (see references at end of this section). The KISS protocol specifies the communication between a TNC and a host (e.g., computer).

KISS Mode allows the TNC to act as a modem and packet assembler/ disassembler (PAD). In KISS Mode, data processing is shifted from the TNC to a computer running special software supporting the KISS code designed by Phil Karn. KISS code supports higher level protocols (i.e. TCP/IP) for sharing computer resources in a network fashion. Examples of TCP/IP programs are NET, NOS, and JNOS. Also, some networking software (G8BPQ) requires the KISS Mode of operation.

Technical Note*: In the KISS Mode, the TNC simply passes all received data to your computer, and the computer program is responsible for all processing of that data, including decisions concerning routing, digipeating, and other control decisions. The TNC converts the packets of synchronous data being received from the radio link into packets of asynchronous data to be passed to the computer over the serial port, and converts the packets of asynchronous data from your computer into the synchronous format suitable for radio transmission. The TNC retains the responsibility of determining proper timing for channel access.*

To place your TNC in KISS Mode, you must first have the command prompt (cmd:). Then type INTFACE KISS and press return. And, finally, do a RESET.

After that you will not see any message or any prompt back from the TNC, it will now be in KISS mode.

If you turn your unit off while it is it INTERFACE KISS mode, it will still be in KISS mode when you turn it back on. To get out of KISS mode, either:

- perform a hard reset, or
- use the TCP/IP command (available in KISS software while in KISS mode), or
- send the C0 FF C0 sequence from your keyboard.

To send the C0 FF C0 sequence from your keyboard (using a PC compatible computer):

1. Press and HOLD the ALT key. Type the numbers 192 from the numeric KEYPAD, not the keyboard. Release the ALT key.
2. Press and HOLD the ALT key. Type the numbers 255 from the numeric KEYPAD, not the keyboard. Release the ALT key.
3. Press and HOLD the ALT key. Type the numbers 192 from the numeric KEYPAD, not the keyboard. Release the ALT key.

If the terminal program you are using sent all those characters, you will be out of the KISS Mode now.

References on KISS Mode

Karn, P.: “TCP/IP: A Proposal for Amateur Packet Radio Levels 3 and 4", pp. 4.62-4.68 ARRL Amateur Radio Computer Networking Conferences 1-4, Newington, CT: ARRL, 1985.

Chepponis, M. and Karn, P.: “The KISS TNC: A Simple Host-to-TNC Communications Protocol”, pp. 38-42, 6th Computer Networking Conference, Newington, CT: ARRL, 1987.

XKISS (Extended KISS) Mode

The TNC also supports the G8BPQ multi-drop KISS Mode, called extended KISS (XKISS). With this mode, the TNC can be connected to a G8BPQ node along with the other TNCs on the same serial port, to provide multiple radio ports to the node. Otherwise, XKISS works like KISS. For information on this mode of operation, see the documentation for the XKISS software you are using.

DAMA (Slave Mode) Capacity

All of Kantronics’ current amateur TNCs (KAM Plus, KPC-3 Plus, and KPC-9612 Plus) support the DAMA protocol (slave mode) as used by many amateur radio operators in Germany, Belgium, Luxembourg, the Netherlands, France, Switzerland and other parts of Europe.

Note: To the best of our knowledge there are no DAMA nodes in operation in the US or anywhere else outside of Europe.

What is DAMA?

DAMA (Demand Assigned Multiple Access) is a network protocol for regulating packet traffic to eliminate or dramatically reduce collisions among users on the crowded channels available to amateurs in densely populated northern Europe. As the European channels have become more and more crowded and more and more DAMA systems have been established, it has become a necessity for TNCs to be equipped with the DAMA slave protocol in order to use the local networks. In the DAMA system, a “master” node is set up in each local area to control transmissions from all stations connected to the node. By local agreement, users do not connect directly to each other; they connect only through the master.

DAMA was developed in Germany by NORD><LINK. (References to more information on DAMA are provided in the bibliography at the end of this section.)

What Does a DAMA System Look Like?

The DAMA master node requires a computer running specialized DAMA software. TheNetNode (TNN) software, developed by NORD><LINK is one such specialized

program which is used in the computer of a DAMA master along with one TNC, with a special EPROM for each radio frequency used on the system. RMNC/FlexNet software is another program, which is very popular in Germany.

Within each Local Area Network (LAN), end users set themselves up as DAMA “slaves,” using specially equipped TNCs, which contain firmware (EPROM) that can operate in a DAMA “slave” mode (such as the Kantronics TNCs with DAMA slave mode support). The slaves may use a standard terminal or host mode program or any of a variety of specially written European programs, such as SP, Grafik Packet and ESKAY, some of which make use of the KISS mode feature of the TNC.

To use the DAMA system, the user connects to the DAMA master with a normal connect frame. When the master responds to the connect frame, the TNC is automatically placed into DAMA slave mode. The DAMA master polls each slave within the LAN in a “round-robin” fashion (somewhat like the token passing method) and the slave TNCs transmit packets when the master has granted permission to do so. The slave TNCs continue to be polled with priority among the slaves assigned according to the number of times each responds with information frames (as opposed to rr frames - i.e., no data to send). Priority is dynamically assigned so that those stations that are sending information are polled more frequently than idle stations. However, once an “idle” station responds with information, its priority is upgraded.

Communication over the wider area is handled between the DAMA master nodes which are connected to each other using a standard NET/ROM™ type protocol. For the DAMA system to work effectively, each slave associates with only one master node and connects only to that master. If an end user is connected to more than one master, *each* master will poll the slave TNC according to its polling schedule. When polled, the slave TNC will transmit *all* frames it has available to whichever master polls it. This would increase the probability of collisions with other users, thus defeating the purpose of the DAMA system.

How is DAMA Implemented in Kantronics’ TNCs?

Most Kantronics TNCs, including the KPC-3 Plus, KPC-9612 Plus, and the KAM Plus, implement the DAMA slave mode, using the following commands (see the Command Reference section for details):

DAMA {ON|OFF}

Set DAMA to ON to have your Kantronics TNC operate as a DAMA slave station once a connection is established with a DAMA master station. When DAMA is OFF, the TNC will operate in its carrier-sense-multiple-access mode (CSMA). For details, see the CD, SLOT, and PERSIST commands in the Command Reference section of this manual.

DAMACHCK n (n = 0 to 225)

Set a “DAMA time-out” timer (in 10-second increments) to specify how long your TNC (running in DAMA slave mode) will wait to be polled by a connected DAMA station before disconnecting from the DAMA station and reverting to CSMA operation.

Bibliography

Bertrem, Eric, F5PJE. “RMNC/FlexNet: The Network of Choice in Western Europe.” In “Packet Perspective” conducted by Stan Horzepa WA1LOU. QST April 1995: 88.

Jost, Gunter, DK7WJ, Sonnabend, Joachim, DG3FBL, and FlexNet-Group. “FlexNet. The European Solution”. *Proceedings of the 9th Computer Networking Conference*. American Radio Relay League, Newington, CT, USA 06111.

Medcalf, Karl, WK5M. “DAMA - Another Network Solution.” *Proceedings of the 14th Computer Networking Conference*. American Radio Relay League, Newington, CT, USA 06111.

Rech, Wolf-Henning, N1EOW/DF9IC and Kneip, Johannes, DG3RBU. “The German (Central European) Packet Radio Network: An Overview.” *Proceedings of the 11th Computer Networking Conference*. American Radio Relay League, Newington, CT, USA 06111.

Schmid, Detlef J., DK4EG. “DAMA - A New Method of Handling Packets?” Trans. Mark Bitterlich, WA3JPY. *Proceedings of the 8th Computer Networking Conference.* American Radio Relay League, Newington, CT, USA 06111. Published in German as “DAMA, ein neues Verfahren für Packet Radio?” *cq*-DL April 1989.

TheNetNode (TNN) Copyright 1994, NORD><LINK e.V., c/o Klaus-Dieter Vieth, Goergesstrasse 3, D-38118 Braunschweig, Germany. NET/ROM is a trademark of Software 2000, Inc.

CAUTION! The DAMA firmware in Kantronics’ TNC is based on documentation published in the ARRL Computer Networking Conference papers (see Bibliography) and documentation provided by NORD><LINK, the developers of DAMA. This firmware was tested with TheNetNode firmware (Version 1.56), TFKISS (Version 2.0) and by volunteer testers in Europe with firmware used in existing DAMA networks. Kantronics cannot be responsible for subsequent changes to the DAMA protocol which may render this firmware obsolete nor guarantee satisfactory operations with newer versions of the protocol and/or complementary software.

Command Reference

Introduction

This chapter documents all KPC-3 Plus commands.

There are many commands which affect operation of the KPC-3 Plus. Some commands affect performance under specific conditions, some change parameters affecting general operation and others direct a one-time action.

The user changes parameters and issues instructions to the KPC-3 Plus by typing commands composed of English-like word abbreviations and variables which are numbers or strings of characters chosen by the user. You will probably never change some of these parameters.

Default values are stored in the EPROM. If you change any setting or value, the new setting or value will be stored in battery-backed RAM and will be the value used at future power-on.

The availability of the commands listed here depends upon what INTFACE (i.e., INTERFACE) mode the KPC-3 Plus is currently in (see the INTFACE command in the command listing below and check the index for more information on INTFACE modes).

Format for Listing Commands

All KPC-3 Plus commands are listed alphabetically below.

A "❖" bullet in front of the name of a command indicates that the command is a NEWUSER Command— it will be available for use when the KPC-3 Plus is set to NEWUSER Mode.

The KPC-3 Plus ships with NEWUSER as the default INTFACE mode. Also, the KPC-3 Plus will be in NEWUSER mode after a hard reset.

Format Defining Commands

The Command Line

The first line of each command definition shows the syntax for entering the command, beginning with the name of the command, followed by a listing of any parameters that may be used with the command.

The full name of each command is shown in CAPS and a "short-form" of the command is shown by the underlined characters in the name. The short-form of the command is the shortest string of characters, starting from the beginning of the full name, that will allow the KPC-3 Plus to uniquely identify the command being requested.

The KPC-3 Plus scans an internal list of commands—which is NOT in alphabetical order—and selects the first match with the string you enter.

The command name is followed by a space.

After the space, there may be one or more parameters that need to be used, or that may optionally be used, with the command.

- Values that need to be entered as they are shown (or by using the short-form indicated by the underlined character(s)) are shown in CAPS , as in “ON | OFF”. Parameter names for which a value needs to be substituted are shown in lower case letters, as in “callsign,” which means a particular callsign needs to be entered.

- Lists of parameter values are enclosed in curly brackets “{ }” for grouping clarity and lists of optional parameter values are enclosed in square brackets “[ ]”.

- If more than one choice is possible for the value of a given parameter (i.e., flag), choices are separated by a vertical bar, as in “ ON | OFF” or “OPTION1 | OPTION2 | OPTION3".

If a parameter can take a range of values (or in cases such as ABAUD, any of a set of numerical values) the parameter is shown as a name in lower case (e.g., “n”) and the permissible range of values (or the list of possible values) is given in parentheses “( )” to the right side of the command, after the syntax has been presented. For example, possible values of the parameter “n” are defined as “(n = 1 - 255)”.

The Line Below the Command Definition

If commands have parameters, default values are shown on the line below the command definition.

Some KPC-3 Plus commands take effect immediately, so they are called “immediate” commands (e.g., K means “switch to CONVERSE mode"). This will be stated on the second line. For example, the command to go to CONVERS mode says “immediate” on the second line.

More Information on Commands

Each command is explained in detail and examples are given where appropriate. Following the explanation of the command, there may be a “**See also**” list that shows related commands.

Parameter Types

Almost all parameters are of one of the following types (exceptions are discussed when they are encountered in the command listing).

n (range)

Any number within the range is permissible. The unit of measure (s, ms, baud, count, etc.) for the number will be given in the description. These are decimal numbers.

n ($00 - $FF)

Special functions: Some parameters are used to control special functions, such as specifying the character to use to enter the Command mode from the Convers mode or specifying the character to use for “backslash”. Possible values and default values for these parameters are shown in HEX format.

The “digits” of a hex number represent powers of 16, analogous to the powers of 10 represented by a decimal number. The decimal numbers 10 through 15 are represented in hex by the letters A through F (e.g., decimal 10 is hex A). And a hex number is distinguished from a decimal number by preceding it with a “$” prefix.

For example:

$1B = (1*16) + 11 = 27

Number codes for these special characters are shown in hexadecimal (hex) form (i.e., base 16). They can be entered either in decimal or in hex. Permissible values are shown in HEX: for example (n = $00 - $FF).

Note: Setting a function character to zero ($00) disables that function.

- See the ASCII Chart at end of this manual for a listing of ASCII codes for all alphanumeric characters, including A to Z and 0 to 9, and conversions between hex and decimal representations.

Several parameters are ASCII numerical values for characters, which perform special functions. Most of these special function characters have “control characters” as default values. Control characters are entered by holding down a special control key (normally marked “Ctrl”) on the keyboard while typing the indicated key. For example, to enter a <Ctrl+X>, hold down the Control key while typing an X, then release both keys.

These special characters cannot be sent in a packet unless preceded by the pass character (see the PASS command) or unless you are operating in the Transparent Mode.

If a streamswitch (STREAMSW) character or any other special character is defined as “$” then you will need to enter values in decimal, or precede the $ with the PASS character in order to enter hex numbers.

flags ChoiceA|ChoiceB

Many parameters are “flags”, meaning they have two possible values, ON and OFF, or YES and NO. All of the command descriptions show ON and OFF as the options; however YES (Y) and NO (N) may be typed instead. A few parameters are really flags,

but rather than indicating that something is “on” or “off”, they select one of two ways of doing things. Some of these parameters have the values EVERY or AFTER indicating operating modes for data transmission.

The possible choices are separated by a vertical bar. Some of the flag parameters will allow many choices, such as ON|OFF|TO|FROM.

callsigns xxxxxx-n

Several commands require callsigns as parameters. While these parameters are normally Amateur callsigns, they may actually be any collection of numbers and/or letters up to six characters; they are used to identify stations sending and receiving packets. A callsign may additionally include an “extension” (SSID, Secondary Station Identifier), which is a decimal number from 0 to 15 used to distinguish two or more stations on the air with the same Amateur call (such as a base station and a repeater). The callsign and extension are entered and displayed as call-ext, e.g. NØGRG-3. If the extension is not entered, it is set to -0, and extensions of -0 are not displayed by the KPC-3 Plus.

text

There are some commands, which have a parameter text string. This string can be any combination of letters, numbers, punctuation, or spaces up to 128 characters.

In order to be used, all string parameters must contain at least one non-space character. You can even put characters with special meanings, such as carriage return, into the string by preceding them with the PASS character. The string ends when you type a (non-passed) carriage return.

Entering Commands

To enter a command, the KPC-3 Plus needs to be in COMMAND Communication Mode (as opposed to CONVERS or TRANS communication mode).

The prompt for Command Mode is:

cmd:

+ Hint: Once you go into Packet Convers Mode a <Ctrl+C> (see COMMAND) needs to be entered to return you to the Command Mode. In the Packet Transparent Mode or GPS Mode, a special sequence is needed to return to Command Mode (see CMDTIME).

+ Hint: If packets have been printing on the screen you may have forgotten which mode you are in. If you wish to see the cmd: prompt type a <Ctrl+C>, to see if you are already in Command Mode. If the prompt does not appear then you are either in Command Mode or in Transparent Mode. If you are in Command Mode, typing a Return will generate the “cmd:” prompt. If you are in Transparent Mode, you will not get the “cmd:” prompt when you press the Return key; instead, you need to enter three <Ctrl+C>s in rapid succession to switch to the Command Mode and get the “cmd:” prompt.

When you are at the Command Mode prompt, you enter a command for the KPC-3 Plus by typing the command name (in upper or lower case) and any required and optional parameter values (argument settings or values).

- The command name and each parameter value must be separated from each other by at least one space.

To ENTER the command press the carriage return “CR” (usually an ENTER key). As soon as the “CR” key is pressed, the KPC-3 Plus will process the command request and take whatever action is specified (or inform you with an “EH?” if it does not understand what you have entered).

As noted above, you can enter the short-form of a command name—the characters underlined in the listing.

The short form is not necessarily the shortest unique string in the name or the first match in alphabetical order—rather it is the first match that the KPC-3 Plus will find as it scans through the ***non-alphabetical listing*** *of commands in its EPROM.*

- **Hint:** You can examine the value of any parameter by typing the command name followed by a “CR”. A special command, DISPLAY, allows you to see the values of all parameters or groups of related parameters.

Commands

All KPC-3 Plus commands are documented below, in alphabetical order. The underlined characters in a command's name show the short-cut version of the command. Required or optional parameters are shown after the command name, as are allowable ranges of values, where appropriate.

8BITCONV {ON | OFF}

default ON

- This command was added to the first TNCs built (1980s) to accommodate terminals that would not receive (or would be confused by) 8-bit characters, only 7-bit ASCII. We've retained the command today just in case you're using such a terminal or computer. Also, it might be that your computer is capable of handling 8-bit characters but your communications terminal program will reject (or be confused or "hung") by them. In these cases, you'll need to set 8BITCONV OFF.

When ON, transmission of 8-bit data is allowed in the packet Convers Mode and Transparent Mode. If you wish to obtain 8-bit data transmission but do not want all the features of Transparent Mode, set this command to ON. If OFF, the 8th data bit is stripped (set to 0) for transmission and all data received will have the 8th bit set to 0. This does not affect the KA-Node, digipeat or PBBS functions.

ABAUD n (n = 0,1200,2400,4800,9600,19200)

default 0

The parameter n sets the baud used for input and output through the serial RS232 port of the KPC-3 Plus to the computer. If 0 is used, the KPC-3 Plus will run an autobaud routine upon power-up. This routine looks for an asterisk (*) character from the attached computer to set the ABAUD parameter.

If you wish to use a different baud, or to perform autobaud the next time the unit is powered up, you must change the ABAUD parameter. If you change the baud in your computer or terminal you should change the baud in the KPC-3 Plus first, then issue the RESET command, then set the new baud on your computer. Otherwise a hard reset will be required to erase the ABAUD setting and reinitialize the KPC-3 Plus to perform the autobaud routine. (See Hard Reset section.) Note also that a hard reset will erase ALL stored parameters in your KPC-3 Plus and return them to factory defaults.

See also: reset, restore

ANALOG

immediate

Entering the ANALOG command returns a string of 8 values, based on the current voltage readings from 8 A-to-D lines. Only the first two of these values are relevant to the user, reporting two external voltages.

At any given time, each of the 8 A-to-D lines will have a voltage value somewhere in the range of 0 - 5 V dc. Instead of reporting voltages, the ANALOG command returns a decimal number in the range of 0 - 255 for each line. Each decimal value corresponds to a voltage input.

The reason ANALOG returns 8 values is that it is attached to an 8-way mux input of a single A/D converter. In the KPC-3 Plus, only the first two values are relevant to the user (several other lines could be used for special purposes, with appropriate modifications of the board, but they are normally intended just for internal use within the KPC-3 Plus).

The response is in the following form:

AN0/AN1/AN2/AN3/AN4/AN5/AN6/AN7

- AN0, input (0), reports a decimal number in the range of (0-255), representing the current dc voltage of an external input read from either pin 4 on the Radio Port or from pin 18 on the Serial Port, depending on the current setting of jumper J8:
  - If jumper J8 is in position 1, the external voltage input is read from pin 4 of the Radio Port.
  - If jumper J8 is in position 2, the external voltage input is read from pin 18 of the Serial Port.

  Note: As shipped, jumper J8 is off to the side, so it is not in either position 1 or position 2. If J8 is in neither position 1 nor position 2, AN0 will not report meaningful values.

- AN1, input (1), reports a decimal number in the range of (0-255), representing the current dc voltage of external input read from either pin 8 on the Radio Port or from pin 11 on the Serial Port, depending on the current setting of jumper J10:
  - If jumper J10 is in position 1, the external voltage input is read from pin 8 of the Radio Port.
  - If jumper J10 is in position 2, the external voltage input is read from pin 11 of the Serial Port.

  Note: As shipped, jumper J10 is off to the side, so it is not in either position 1 or position 2. If J10 is in neither position 1 nor position 2, AN1 will not report meaningful values.

- AN2, is not used in the KPC-3 Plus.

- AN3, is not used in the KPC-3 Plus.

- AN4, input (4), is intended for internal use.

- AN5, is not used in the KPC-3 Plus.

- ♦ AN6, input (6), is intended for internal use. It reports the current status of the RTS pin for the serial port, which is either 0 (or near 0) for logic low or 255 (or near 255) for logic high.

- ♦ AN7, input (7), is intended for internal use. It reports the current status of the DTR pin for the serial port, which is either 0 (or near 0) for logic low or 255 (or near 255) for logic high.

- ● Data can be gathered from a remote site by connecting to MYREMOTE and then issuing the ANALOG command.

See also: range, telemetry, myremote

AUTOLF {ON | OFF}

default ON

When ON, a line feed is sent to the terminal after each carriage return. This parameter should be set on when overprinting occurs and the terminal being used does not automatically supply its own linefeed after a carriage return. This command affects only the data sent to the terminal, not data sent to the radio.

See also: cr, lfadd

AX25L2V2 {ON | OFF}

default ON

This command provides compatibility with all known packet units implementing AX.25 protocol. When ON, Level 2 Version 2 protocol is implemented and the TNC will automatically adapt to whichever version the connecting station is using. When OFF, Level 2 Version 1 is implemented. Set this command to OFF if you need to digipeat through other units, which do not digipeat version 2 packets. You may also find benefit from setting this command OFF when using several digipeaters (not nodes) to send packets, or when conditions are marginal between the two stations involved. (NOTE: Changing this setting after connecting to another station will have no effect on the current connection.)

The major difference in V1 and V2 protocol is the method used to handle retries. In the connected mode, if a packet is sent and not acknowledged, Version 1 will re-send the entire packet and then disconnect if the RETRY count is reached. Version 2 will first send a poll, the response to this poll will determine if the packet was received. It is possible that the ack was collided with and therefore the data packet does not need to be resent. If the ACK shows that the data packet was not received, it will be re-transmitted. Each time a poll is answered the TRIES count is reset to 0. If the RETRY count is reached, version 2 will attempt to re-connect unless RELINK is OFF. If the re-connect attempt is unsuccessful, then version 2 will issue a disconnect.

See also: relink, retry, tries

For more information, see the book *AX.25 Amateur Packet-Radio Link-Layer Protocol Version 2.0 October 1984*, which can be obtained from the ARRL.

AXDELAY n **(n = 0 - 255)**

default 0

The value of “n” is the number of 10 millisecond intervals to wait, in addition to TXDELAY, after keying the transmitter before data is sent. This delay can be helpful when operating packet through a standard “voice” repeater, or when using an external linear amplifier which requires extra key-up time. Repeaters using slow mechanical relays, split-sites, or both require some amount of time to get RF on the air.

See also: axhang

AXHANG n **(n = 0 - 255)**

default 0

Each increment specifies 10 millisecond intervals. This value may be used to improve channel utilization when audio repeaters with a hang time greater than 10 ms are used. If the repeater squelch tail is long, it is not necessary to wait for AXDELAY after keying the transmitter if the repeater is still transmitting. If the TNC has heard a packet within the AXHANG period, it will not add AXDELAY to the key-up time.

See also: axdelay

BEACON [EVERY | AFTER] n **(n = 0 - 255)**

default Every 0

The value entered for “n” is the number of minutes (i.e., one-minute intervals) to use with the beacon command.

A value of 0 turns the beacon OFF. Setting a value greater than 0 activates the beacon under the conditions specified. If the optional keyword Every is used, a beacon packet will be sent every n minutes. If set to After, a beacon packet will be sent ONCE after the specified interval with no channel activity.

The beacon frame consists of the text specified by BTEXT in a packet addressed to “BEACON”. Beacon messages will be digipeated via any addresses specified in the UNPROTO command.

See also: btext

❖ **BKONDEL {ON | OFF}**

default ON

When ON, the sequence backspace-space-backspace is sent to the attached terminal when the DELETE character is entered. When OFF, the backslash character "\" is sent to the terminal when the DELETE character is entered.

See also: delete, redisplay

BLT n {EVERY | AFTER} hh:mm:ss [START hh:mm:ss] [CLEAR] **(n = 1 - 4)**

default EVERY 00:00:00 (for each of the 4 LT strings)

This command (used with GPS) sets the interval between beacons for the associated LT (Location Text) string.

n identifies which LT is used (1-4). If EVERY is specified, the LT will be transmitted at the interval specified by the hh:mm:ss parameter. This is set in hours:minutes:seconds, and all characters must be specified. If the optional key word START is given followed by a time (in 24 hour format), the beacons will not be transmitted until the specified time. The beacon will then be transmitted according to the setting of the EVERY parameter. (NOTE: If the current time is past the start time, the beacon will start at the next scheduled interval based on the start time.)

If AFTER is specified, the beacon will be transmitted once after no activity is detected on the channel for hh:mm:ss time. Selecting AFTER does not allow the optional START parameter.

For example, setting BLT 3 EVERY 00:15:00 START 03:30:00 would cause the third LT string (LT 3) to be transmitted every 15 minutes, with the first transmission each day occurring at 03:30:00.

If CLEAR is specified, the LT buffer is cleared after transmission. This way no copies of a NMEA string are repeated.

BREAK {ON | OFF}

default OFF

If BREAK is ON, a break at the keyboard causes a return to Command Mode from Converse or Transparent Mode.

See also: COMMAND

BTEXT text **(0 - 128 characters)**

default (blank)

BTEXT specifies the content of the data portion of the beacon packet. Any combination of characters and spaces may be used with a maximum length of 128.

Entering a single "%" will clear BTEXT.

See also: beacon

BUDLIST [ON|OFF] [NONE | {+|-}call | call1,call2,...]

where call={callsign | {<|>}callsign | callsign{>|<>}callsign}

default OFF NONE

BUDLIST is used to determine which received packets will be monitored. When OFF or NONE, BUDLIST will allow monitoring of all packets, even if it has callsigns in its list.

The maximum number of callsigns allowed in BUDLIST is 10. Up to 10 callsigns may be entered at once, using the format shown after the command name, and the new list of from 1 to 10 callsigns will replace whatever list was there before. The BUDLIST command can be used to add a single callsign (+callsign), so long as there is room for the new callsign on the list (if there is not, the TNC responds with “EH?” and returns you to the command prompt). Similarly, you can remove a single callsign (-callsign) from the list. And to remove all items from the current list, enter BUDLIST NONE.

A callsign entered without a SSID will match any SSID of that callsign. To match a specific SSID only, enter the callsign with that SSID: for example, callsign-n, where n = 0-15.

More selective monitoring is enabled when a callsign or two callsigns is combined with a “>” and/or a “<” character. For example, entering “>callsign” will enable monitoring of packets TO that callsign (assuming BUDLIST is ON), entering “<callsign” will enable monitoring of packets from the callsign, entering “callsign1>callsign2" will enable monitoring of packets from callsign1 to callsign2, and entering ”callsign1<>callsign2" will enable monitoring of packets from callsign1 to callsign2 and vice versa.

- Note that entering callsign1>callsign2 or callsign1<>callsign2 counts as two of the 10 maximum allowed callsigns.

CALIBRAT

immediate

The CALIBRATE command generates a signal which may be used as an aid in adjusting the audio drive level of the modem connected to your transceiver.

The use of this command is explained in the Calibration/Equalization Section.

See also: Calibration/Equalization section

CANLINE n **(n = $00 - $FF)**

default $18 <Ctrl+X>

This command is used to change the cancel-line input editing command character. When in Convers or Command Mode entering a <Ctrl+X> will cancel all characters input from the keyboard back to the last un-PASSed carriage return (unless PACTIME has expired, CPACTIME is turned on, and you are in Convers Mode).

See also: canpac, cpactime, pass

CANPAC n **(n = $00 - $FF)**

default $19 <Ctrl+Y>

This command is used to change the cancel-packet command character. When in the Convers Mode entering a <Ctrl+Y> will cancel all keyboard input back to the last unpassed SENDPAC character (unless PACTIME has expired and CPACTIME is turned on).

This character also functions as a cancel-output character in Command Mode. Typing the cancel-output character a second time re-enables normal output. For example, if you've told the TNC to do a DISPLAY, a <Ctrl+Y> will cancel the display and a second one re-enables the cmd: prompt after the next "CR".

See also: canline, cpactime, sendpac

CD {INTERNAL | EXTERNAL | SOFTWARE}

default INTERNAL

The CD command selects which carrier detect method will be used.

When set to INTERNAL, the TNC will detect a signal present on the channel, using an energy type carrier detect, allowing shared voice and data on the same channel.

When set to EXTERNAL, the carrier detect is supplied by an external device, connected to the XCD pin on the radio port.

If set to SOFTWARE, the firmware inside the TNC will detect the presence of data to enable the carrier detection, allowing operation with un-squelched audio. Correct operation of SOFTWARE carrier detect is affected by proper equalization and the SWP parameter. If your RCV light flickers, this is an indication that you may need to adjust the equalization. (Equalization is set with an internal jumper.)

When CD is set to INTERNAL or SOFTWARE, the external carrier detect pin on the radio connector (if wired) is used to hold off the TNC from keying the radio.

- Hint: For one use of this function, see the APRS and GPS section of this manual.

See also: swp

CHECK n **(n = 0 - 255)**

default 0

This “timeout” command is used to prevent a “hang-up” in a connected mode when a link failure occurs as a result of conditions beyond control of the connected stations. The value entered for “n” is the number of 10 second intervals to wait during “no activity” before making a periodic check (poll) to determine that a connected state still exists.

If n equals 0, this “timeout” function is disabled. While using Version 1 (AX25L2V2 OFF), a check timeout will initiate a disconnect.

See also: ax25l2v2, kntimer, relink, rnrtime

CMDTIME n **(n = 0 - 15)**

default 1

This command sets the time allowed for entry of required characters to exit the Transparent Mode. In order to allow exit to Command Mode from Transparent Mode, while permitting any character to be sent as data, a guard time of CMDTIME seconds is set up.

Entering a value for “n” sets the value of CMDTIME to “n seconds”.

To exit Transparent mode, you need to wait at least CMDTIME since the last data character was sent to the KPC-3 Plus. Then you need to enter the COMMAND character (e.g., <Ctrl+C>)) three times, with a wait of LESS THAN the value of CMDTIME between the first and second time you enter the COMMAND character and also a wait of LESS THAN the value of CMDTIME between the second and third time you enter the COMMAND character.

After a final delay of CMDTIME the TNC will exit Transparent Mode and enter Command Mode. At this time you should see the cmd: prompt. If CMDTIME is set to zero, the only exit from Transparent Mode is a modem break signal, assuming BREAK is set to ON.

For example (if CMDTIME is 1 second and COMMAND is <Ctrl+C>): wait one second, type a <Ctrl+C>, within one second type a second <Ctrl+C>, within one second type a third <Ctrl+C>, WAIT one second, cmd: prompt should appear. If your computer/program has the capability you can also send a modem break to escape Transparent Mode.

See also: command, trans, break

CMSG {ON | OFF | DISC | PBBS}

default OFF

When OFF, the custom connect text stored in CTEXT will not be sent to the connecting station upon receiving a connect request. When ON, the custom CTEXT string will be sent. When CMSG is set to DISC, the custom CTEXT string will be sent to the connecting station, and then your TNC will disconnect from that station. If set to PBBS, the custom CTEXT string will be sent to the connecting station, and then the connection will automatically be transferred to your PBBS. If the PBBS is not available, your TNC will disconnect from the station.

See also: ctext, pbbs

COMMAND n **(n = $00 - $FF)**

default $03 <Ctrl+C>

This command is used to change the Command Mode entry character. When COMMAND is set to the default value, typing a <Ctrl+C> causes the TNC to return to Command Mode from packet Convers Mode. See CMDTIME for returning to Command Mode from Transparent Mode.

CONLIST [ON | OFF] [NONE | {+|-}callsign | callsign1,callsign2...]

default OFF

CONLIST is used to determine which stations (callsigns) may use your station for ANY purpose, including digipeating.. When ON, the TNC will recognize only those packets received with a callsign that appears in the CONLIST's list of callsigns; and you will not be able to connect to any station that is not in the conlist. When OFF, the TNC will process all packets.

The maximum number of callsigns allowed in CONLIST is 10. Up to 10 callsigns may be entered at once, using the format shown after the command name, and the new list of from 1 to 10 callsigns will replace whatever list was there before. The CONLIST command can be used to add a single callsign (+callsign), so long as there is room for the new callsign on the list (if there is not, the TNC responds with "EH?" and returns you to the command prompt). Similarly, you can remove a single callsign (-callsign) from the list. And to remove all items from the current list, enter CONLIST NONE.

A callsign entered without a SSID will match any SSID of that callsign. To match a specific SSID only, enter the callsign with that SSID: for example, callsign-n, where n = 0-15.

CONMODE {CONVERS | TRANS}

default CONVERS

This command controls the mode the TNC will be placed in AUTOMATICALLY after a connect if NOMODE is OFF. The connect may result either from a connect request received or a connect request originated by a CONNECT command. If the TNC is

already in Convers 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 input.

See also: canline, connect, convers, nomode, trans

❖ **CONNECT call1 [VIA call2,call3,...,call9]**

immediate

call1 = callsign of station to be connected to.

call2,...,call9 = optional stations to be digipeated through. A maximum of 8 digipeater addresses (callsigns or aliases) can be specified. This is referred to as a path.

Each callsign may also have an optional Secondary Station Identifier (SSID) specified as -n, where n = 1 - 15. The digipeat callsigns are specified in the order in which they are to relay transmitted packets. The mode set by CONMODE will be entered upon successful connect, if NOMODE is OFF. If no response to the Connect request occurs after RETRY attempts, the command is aborted. A timeout message is printed on the display and the TNC remains in the Command Mode. The station being connected to (call1) may receive the connect request but be unable to accept connects, in which case a busy message will be printed to the screen and the TNC will stay in Command Mode. Connect requests may only be initiated in the Command Mode and the connect will be established on the current stream.

If a connect is in progress, or already established, the path may be changed by simply re-issuing the CONNECT command with the desired path. This must be done on the same stream as the original connect.

CAUTION, packets en route between your station and the reconnected station may be lost.

If CONNECT is entered with no parameters, the status of the current stream is displayed.

See also: conmode, conok, dbldisc, maxusers, nomode, retry, ring, streamsw, xmitok

CONOK {ON | OFF}

default ON

When ON, connect requests from other TNCs will be automatically acknowledged and a <UA> packet will be sent. The standard connect message, with stream ID if appropriate, will be output to the terminal and the mode specified by CONMODE will be entered on the I/O stream if you are not connected to another station and NOMODE is OFF.

When OFF, connect requests from other TNCs will not be acknowledged and a <DM> packet will be sent to the requesting station. The message “connect request: (callsign of whoever is trying to connect to you)” will be output to your terminal if INTFACE is TERMINAL or NEWUSER.

When CONOK is OFF, you can still connect to your mailbox.

When operating with multiple connects allowed, the connection will take place on the next available stream. Connect requests in excess of the number allowed by the USERS command will receive a <DM> response and the “connect request: (call)” message will be output to your terminal if INTFACE is TERMINAL or NEWUSER.

See also: conmode, connect, intface, maxusers, monitor, nomode, users

CONPERM {ON | OFF}

default OFF

Setting CONPERM to ON forces the connection on the current stream to become permanent and causes the KPC-3 Plus to attempt to reconnect when it is restarted. CONPERMED connections are shown with /P in the status display, which is generated by the STAT command.

See also: status

❖ **CONVERS**

immediate

CONVERS has no options. It is an immediate command and will cause entry into Conversational Mode from Command Mode on the current I/O stream. Any link connections are not affected.

- Hint: K is the same as CONVERS, for quicker entry.

See also: k, command

CPACTIME {ON | OFF}

default OFF

When OFF and in the Convers Mode, packets are sent when the SENDPAC character is entered or when PACLEN is achieved. When ON and in the Convers Mode, packets are sent at periodic intervals determined by PACTIME. Characters are sent periodically as in Transparent Mode but the local editing and echoing features of Convers Mode are enabled.

See also: convers, cr, paclen, pactime, sendpac, trans

CR {ON | OFF}

default ON

When ON the SENDPAC character (normally carriage return) is appended to all packets sent in Convers Mode except when PACLEN is exceeded. 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 the “CR” appended to the end of the line. To avoid overprinting, AUTOLF may need to be ON at the receiving end.

See also: autolf, lfadd, sendpac

CRSUP {ON | OFF}

default OFF

This command was added to the first multi-mode TNCs (such as the Kantronics UTU and KAM) to accommodate the practice by radio teletype (RTTY) operators of adding an extra carriage return (CR) at the end of each line (i.e., CR, CR, linefeed (LF)). This was done to give the carriage of the old mechanical teletypes time to return across the page. Some of these machines are still in use, so we’ve retained this command so you can “correct” the data you might receive from such a station. To do so, turn CRSUP ON. When ON, this command suppresses every other carriage return (when no data is between them) in the received data before sending the data to the terminal. When CRSUP is OFF, all carriage return characters are left in the received data as it is sent to the terminal.

See also: autolf, lfsup

CSTAMP {ON | OFF}

default OFF

When ON, the daytime stamp is printed with all “*** CONNECTED TO” and “*** DISCONNECTED” messages on the terminal.

See also: connect, daytime, disconnect, mstamp

CTEXT text (0 - 128 characters)

default (blank)

Enter any combination of characters and spaces up to maximum length of 128. Entering a single ‘%’ will clear CTEXT. This entry specifies the text of the first packet to be sent in response to an accepted connect request provided that the parameter CMSG is not OFF.

See also: cmsg, connect

CTRL [A | B] {n | ON | OFF | LONG | MUTE m} **(n=1-20)**

immediate

This command causes the KPC-3 Plus to activate the specified output line (A or B) to the radio port as indicated. If A or B output is not specified, both outputs are affected. For instance, to pulse the A output (CTLA on the schematic) 3 times, you would give the command “CTRL A 3". You may optionally specify the number of times to pulse the output (n), or to turn the output ON, or OFF. Each pulse on the output line lasts approximately 100 ms (0.1 s). Specifying LONG results in one pulse of the output line, lasting approximately 1.5 s. These output lines are open drain outputs and are pulled to ground when active. For more information, see the section on “Connecting to your Radio” in this manual.

With MUTE, the control line can be used to mute voice repeater outputs. When MUTE is specified, the output control line will be a filtered version of the RCV led, with about a 10*m ms attack time. If the RCV led is active for m consecutive samples at 10 ms intervals, the CTRL line will activate; if the RCV led is off at any sample, the CTRL line will open.

Do not allow more than 200 mA of current in this circuit.

CWID [EVERY | AFTER] n **(n = 0 - 255)**

default EVERY 0

Each increment specifies 1-minute intervals. A value of 0 turns the ID OFF. Setting a value greater than 0 activates the ID under the conditions specified. If the optional keyword “Every” is used, an ID will be sent every n minutes. If set to “After”, an ID will be sent ONCE after the specified interval with no channel activity.

The string specified by the CWIDTEXT command will be sent in Morse Code, using AFSK tones.

Some countries require all stations to ID in Morse code periodically.

See also: mycall, cwidtext

CWIDTEXT text **(0-15 characters)**

default DE mycall

This command sets the text to be transmitted when the KPC-3 Plus performs an automatic CWID. The text will be transmitted periodically as set by the CWID command.

See also: cwid

DAMA {ON | OFF}

default OFF

When ON the TNC can operate as a DAMA slave station once connection is established with a DAMA master station. When OFF the TNC operates in standard Packet mode (i.e., CSMA—Carrier Sense Multiple Access).

See also: damachck

DAMACHCK n **(n = 0 to 255)**

default 18

This command sets a DAMA timeout timer in 10-second increments. If the TNC is connected to a DAMA master station (that is, operating in DAMA slave mode) and the master does not poll the TNC for data before the DAMACHCK timer expires, the TNC will revert to CSMA operation and disconnect from the DAMA master.

This command was added to allow the user to regain CSMA connect capability in the event the DAMA master fails to poll because of a node failure – lightning strike, power outage, etc.

See also: dama

DAYSTR text

default mm/dd/yy hh:mm:ss

The DAYSTR command is used to set the FORMAT of your date/time display. DO NOT enter an actual date or time, simply enter the form of the display you would like, using the lower case letters m, d, h, y, and s as described below.

The format you enter is used for all time stamps, including the PBBS, KA-Node, Mheard list, etc.

The lower case characters m, d, y, h, and s have special meaning to this command and will be replaced with data from the software clock. The lower case m will be replaced with the minutes the first time it appears after a lower case h. If h, m, y, d, or s is specified as a single character, the corresponding date/time element will be displayed as a single digit if the value is less than 10. Entering two characters will force a two digit display for values under 10. If the month is entered as three ms, it will be displayed as the first three characters of the month name (JUL). You may also enter any other text you wish, allowing you to add such things as your time zone.

- Remember ALL lower case m, d, y, h, and s will be replaced by actual values.

Some examples of setting the DAYSTR command and the resulting time/date display are:

DAYSTR setting	Displayed Date/Time
mm/dd/yy hh:mm:ss	07/16/02 12:14:22
d.m.y h:mm:ss	16.7.02 12:14:22
d.mm.yy h:mm	16.07.02 12:14
mmm d 20yy h:mm CST	JUL 7 2002 12:14 CST
TIME hh:mm DATE: mmm dd, 20yy	TIME 12:14 DATE JUL 07, 2002

- This command is used to define how date and time are to be displayed; it is not used to enter an actual date and time (which you do by using DAYTIME). If you enter an actual date and time with the DAYSTR command, your clock will appear to stop.

See also: daytime

❖ **DAYTIME yymmddhhmm[ss]**

DAYTIME is used to enter a date and time (after which time and date are formatted for display by the specifications given in DAYSTR).

When date and time are entered, using the input format specified in the command and explained below, the values entered are used to set the software clock/calendar. The software clock/calendar is then used for MHEARD and NDHEARD logging, as well as for CSTAMP and MSTAMP functions.

When entering the daytime digits, enter them in pure number sequence with no spaces, dashes, or slashes. Notice that there are two digits each for entering a year, month, day, hour, minute, and (optionally) second. Also, notice that you need to use the order given here, which has nothing to do with the format (given in DATSTR) for displaying date and time values. For example, to enter 1986, January 2, at 22:30:00 hours, the value of the DAYTIME parameter would be 860102223000. In this case, seconds were entered, so the string is 12 characters long (2 characters for each of six pieces of information entered).

If DAYTIME is entered with no parameter, the daytime is displayed using the format defined by the DAYSTR command.

- Hint: If the time appears to be stuck, check the current value of DAYSTR, to be sure that the DAYSTR setting is being used to format the display of time and date, and not (in error) to attempt to supply an actual date and time. If this is not the case, fix the problem and see if time and date displays work correctly.

See also: cstamp, daytweak, daystr, mheard, mstamp

DAYTWEAK n **(n = 0 - 15)**

default 8

This parameter is used to adjust the software clock, if needed, for accurate time keeping. Increasing the parameter will slow the clock, decreasing the parameter will

speed up the clock. Each count corresponds to 0.64 seconds increase or decrease per 24 hours. Ambient temperature will affect the clock to some degree.

DAYTWEAK has no effect on the (optional) battery-backed clock (BBC); that clock is set only by using the DAYTIME command. The software clock stops when a soft reset or powering off occurs, and upon power on the software clock is set to the value of the BBC, if installed.

See also: wefax

DBLDISC {ON | OFF}

default OFF

When OFF, only one disconnect command (D) need be given to terminate an unsuccessful connect attempt. If you are actually connected, the normal disconnect sequence will occur.

When ON, a normal disconnect sequence will always occur (you will not be disconnected until you receive an acknowledge of your disconnect or until the retry count is exceeded). A second D is required to force a local disconnect independent of the retry counter.

See also: disconnect

DELETE n **(n = $00 - $FF)**

default $08 <Ctrl+H>

This command sets the character to be used as the delete character. When this character is typed, the last input character is deleted. The most common settings are $08 (backspace) and $7F (delete).

See also: bkondel

DIGIPEAT {ON | OFF | UIONLY}

default ON

When ON, any packet received that has MYCALL, MYALIAS, or MYNODE (if KA-Node is active) in the digipeat list of its address field will be retransmitted. Each station included in the digipeat list relays the packet in the order specified in the address field. Digipeating takes place concurrently with other TNC operations and does not interfere with normal connected operation of the station. To disable digipeat operations (via MYCALL, MYALIAS, or MYNODE) turn this command OFF.

If UIONLY is selected, the KPC-3 Plus will digipeat UI frames only.

See also: hid, myalias, mycall, mynode

❖ **DISCONNE [MYPBBS | MYNODE x]** **(x=KA-Node circuit)**

immediate

When given without options, “disconnect” will initiate an immediate disconnect request on the current I/O stream. A successful disconnect results in the display of “*** DISCONNECTED”. If the RETRY count is exceeded while waiting for the connected station to acknowledge, the TNC moves to the disconnected state on that stream. Entering a second Disconnect command before RETRY has expired will result in an immediate disconnect on your end, but may leave the other station thinking it is still connected to you. Disconnect messages are not displayed when the TNC is in Transparent Mode. Other commands may be entered while the disconnect is in progress.

DISCONNE MYPBBS

Issue this command if you want to cause the personal mailbox to issue a disconnect to the user of the mailbox. D MYPBBS is what you should type, do not type the call entered in the mypbbs command.

DISCONNE MYNODE x **(x = KA-Node circuit)**

x may be any of the KA-Node circuits in use, designated by A, B, C, etc. This command will cause the node to disconnect the stations linked through the node on the circuit specified. MYNODE does not refer to the call entered in the mynode command, but is the actual characters to type.

See also: dbldisc, newmode, retry, status

❖ **DISPLAY [{ A | C | G | I | L | M | N | P | T }]**

immediate

This command causes the TNC to display a list of all the parameters in the TNC. You may also display only a selected group of parameters by specifying the appropriate class identifier for that group. When using the DISPLAY command with a subclass be sure to use a space between the DISPLAY command and the subclass. Subclasses of related parameters are:

ASYNC	asynchronous port parameters (TNC to computer)
CHAR	special TNC characters
GPS	parameters related to GPS operations
ID	ID parameters
LINK	parameters affecting packet link (TNC to TNC)
MONITOR	monitor parameters
NET	parameters affecting K-Net
PBBS	mailbox commands
TIMING	timing parameters

Individual parameter values can be displayed by entering DISPLAY and the command name, followed by “CR”.

See also: Display Listings section

❖ **DWAIT n** **(n = 0 - 255)**

default 0

DWAIT defines a delay to be used to avoid collisions with digipeated packets. The value entered for “n” is the number of 10 ms intervals, after last hearing data on the channel, for the KPC-3 Plus to wait before it begins its own key-up sequence.

This value should be established and agreed on by all members of a local area network. The best value will be determined by experimentation but will be a function of the key-up time (TXDELAY). This feature is made available to help alleviate the drastic reduction of throughput, which occurs on a channel when digipeated packets suffer collisions. Digipeated packets are not retried by the digipeater but must be restarted by the originating station. If all stations specify DWAIT, and the right value is chosen, the digipeater will capture the frequency every time it has data to send since digipeated packets are sent without this delay.

Observations have proven that a better algorithm for avoiding collisions between end-user stations, while still allowing digipeaters the high-priority access they require is achieved using Persistence and Slottime to determine proper transmit intervals, and setting DWAIT to 0.

See also: persist, slottime

❖ **ECHO {ON | OFF}**

default ON

When ON, characters received from the computer by the TNC are echoed back and displayed. If you are receiving double print of characters entered at the keyboard, turn this command OFF. This corresponds to the setting in your terminal program for duplex. If your program is set for full-duplex set ECHO ON. If your program is set for half-duplex (some call it echo) then set ECHO in the TNC to OFF. Regardless of the setting of this command, the TNC will not echo an X-OFF or X-ON character to the terminal when it receives a STOP or START character. Echo is disabled in Transparent Mode.

See also: bkondel, flow

ESCAPE {ON | OFF}

default OFF

This command specifies the character, which will be sent to the terminal when an escape character ($1B) is received in a packet. When OFF, $1B is sent. This is useful if your terminal interprets ESC characters as screen positioning commands (ANSI). When ON, the escape character is sent as a dollar sign ($).

FILTER {ON | OFF}

default OFF

When ON, this command will inhibit the printing of control characters (hex $00 - $1F) which may be present in monitored packets. This will be useful if you are monitoring channel traffic, which includes binary file transfers, or higher level protocols (networks talking to each other). Control characters, which may be embedded in those packets, can have strange and unpredictable effects on the monitoring TNC. All control characters except carriage return ($0D) and line feed ($0A) will be filtered. This command DOES NOT affect receipt of control characters in packets received from a "connected" station when MONITOR or MCON is OFF.

See also: monitor

FLOW {ON | OFF}

default ON

When FLOW is ON, any character entered from the terminal will halt output to the terminal until the current packet or command is completed (by SENDPAC, PACLEN, or PACTIME). Canceling the current input to the TNC or typing the REDISPLAY-line character will also cause output to resume. FLOW will keep the display of received data from interfering with data entry.

When FLOW is OFF, received data will be "interleaved" with keyboard entry. If using a split screen terminal program, you should have FLOW OFF and ECHO OFF to allow received data to be displayed while you type into the TNC's type-ahead buffer.

See also: canline, canpac, cpactime, echo, paclen, redisplay, sendpac

FRACK n **(n = 1 - 15)**

default 4

Each increment specifies 1-second intervals. After transmitting a packet requiring acknowledgment, the TNC waits FRACK seconds before incrementing the retry counter and sending the packet again. If the retry count (specified by the RETRY command) is exceeded, the current operation is aborted. If the packet address includes digipeaters, the time between retries is adjusted to FRACK * ((2 * m) + 1) where m is the number of digipeater stations specified. When the retried packet is sent, a random wait time is also added to avoid lockups where two units repeatedly collide with each other.

The FRACK timer begins when PTT is released (the packet has been sent) and is suspended when data carrier from the radio is present, or when your station is transmitting.

See also: connect, resptime, retry

FULLDUP {ON | OFF | LOOPBACK}

default OFF

When OFF (i.e., half duplex), the data carrier detect signal is used as a packet collision avoidance signal. When ON, the modem is run full duplex, and carrier detect does not inhibit transmission. The full duplex mode may be useful especially for satellite operations using duplex radio setups. Full duplex should not be used unless both you and the station you are communicating with have full duplex capability. LOOPBACK is the same as half duplex (i.e., FULLDUP OFF) as far as protocol is concerned, but the receive circuit is still active; this allows you to connect a simple wire between the transmit and receive pins on the radio connector to verify operation of the modem circuitry.

If you have the MX version of the KPC-3 Plus, you will not be able to specify ON or LOOPBACK unless you have installed a second MX modem chip.

In the most recent SMT revision of the KPC-3 Plus circuit board, there is no option for a second modem chip. Therefore, OFF is the only setting for this command.

GPSHEAD n string (n = 1-4) (string up to 8 chars)

default 1 $GPGGA (string 2, 3, and 4 = <blank>)

This command determines which GPS NMEA sentences will be stored in the LT buffers. n (1-4) determines which buffer will be used to store the data, and string is a NMEA sentence header.

When the header is received from the GPS unit, the KPC-3 Plus will store up to 128 characters in the associated LT buffer.

- To clear one of the four strings, enter the buffer number, n (n = 1-4), and nothing else, then press the ENTER key.

See also: blt, lt, ltp, ltrack

GPSINIT string (string up to 128 characters)

default (blank)

This command establishes a string which will be sent to the attached GPS unit upon power-up (i.e., initial text sent to terminal in GPS Mode). This may be useful to configure your GPS unit to provide only the information you require. Some GPS units may require more than one sentence; consult your GPS unit manual. To send more than one, enter a <Ctrl+N> at the end of each sentence, and the KPC-3 Plus will send a CR/LF sequence to the GPS unit. A single % will clear the string.

See also: gpshead, intface

GPSPORT [baud] [NORMAL | INVERTED] [CHECKSUM | NOCHECK]

default 0 NORMAL CHECKSUM

This command allows for a second serial input port for a GPS unit, using a firmware UART.

If ‘baud’ is non-zero, the XCD pin in the radio port can be used as a GPS input during regular operation of the KPC-3 Plus. ‘baud’ may be set to 0, 300, 600, 1200, 2400, or 4800. If NORMAL is specified, the sense (polarity) of the XCD pin is RS232; the XCD pin would be expecting RS232 input or 0-5 V input of the same polarity. If INVERTED is specified, the XCD pin would be expecting polarity opposite of RS232. Most stand-alone GPS units use RS232 or 0-5 V output of the same polarity. If CHECKSUM is specified, the firmware will pass along only GPS sentences, which have valid checksums, and also GPS sentences, which do not have a checksum. Not all GPS units will append checksums to the GPS sentences that they send; some GPS sentences require a checksum, while others do not.

If you wish to use the GPSPORT, note that the KPC-3 Plus cannot send any initialization string to the GPS unit; you must pre-initialize the GPS unit before connecting it up to the XCD pin.

Note: The GPSPORT is a firmware UART. Although the KPC-3 Plus was tested and found to accept GPS data without error at 4800 bps while the KPC-3 Plus was idle, errors occurred if the KPC-3 Plus was active sending and receiving packets and communicating over the RS232 port. Tests found that few, if any, errors occurred when ‘baud’ was set to 2400. For best results, set ‘baud’ as low as possible for your application; if you need to set ‘baud’ to 4800, you should also use the CHECKSUM parameter to discard errors.

Note: When GPSPORT is zero (0), the XCD pin reverts to its original function, inhibiting PTT. If the INTFACE command is set to GPS, the TNC assumes that the GPS device is attached to the normal RS232 port and not the secondary serial input. If INTFACE is set to KISS, XKISS, GPS, or MODEM, you should set GPSPORT to zero.

GPSTIME { OFF | VALID | ON } { GGA | GLL | RMC | ZDA }

default VALID RMC

If GPSTIME is set to OFF, the TNC software clock is not set when NMEA strings are received. If set to VALID, the clock is set but only if the valid bit in the string specified is set. If set to ON, the time is acquired from the string specified.

Note:

- In any event, any string must have a valid checksum.
- Time is not updated if the time received in a string is the same as the last time received.

- GGA and GLL strings do not include a date.
- The ZDA string does not include a “valid” bit.
- Some GPS units do not include a “valid” bit with the GLL string.

HBAUD n **(n = 300, 400, 600, 1200)**

default 1200

This baud specifies the rate of data exchange between the radio stations. The value of HBAUD has NO relationship to the terminal baud specified with ABAUD. In order to communicate with other packet stations, the baud must be the same at each end of the link. As a general rule, 300 baud is used on frequencies below 28 MHz, 1200 baud is used on frequencies above 28 MHz. FCC rules currently limit the maximum baud to 300 when operating below 28 MHz. Although the KPC-3 Plus can operate 300 baud, the modem tones will still use an 800 Hz shift whereas most HF packet uses a 200 Hz shift.

HEADERLN {ON | OFF}

default ON

When ON a carriage return is output to the terminal between the header and text of monitored packets. This causes the packet header and time stamp (if on) to be displayed on one line, with the packet text displayed below it on the next line. When receiving packets addressed only to you (MONITOR and/or MCON OFF) this parameter does not apply. When OFF the data will be on the same line as the header.

See also: cstamp, mcon, monitor, mstamp

❖ **HELP [command]**

immediate

Entering the word “HELP” alone, without any argument, will generate a display listing all commands available in the KPC-3 Plus.

When the name of a command is entered also (e.g., HELP CONVERS), a brief description of that command will be displayed. A wild card also can be used; entering “Help C*” will display brief descriptions of all commands beginning with C.

HELP can also be called by entering “?”.

HID {ON | OFF}

default ON

When ON, an ID packet will be sent every 9.5 minutes, provided that packets are being digipeated through your station, or routed through your KA-Node, or into your PBBS. This command should be ON if digipeating, KA-Node or pbbs is enabled. If OFF, periodic identification packets will not be sent.

See also: digipeat, id, myalias, mynode, mypbbs, numnodes, pbbs

HTEXT text

default blank

The HTEXT command is used by your PBBS to provide hierarchical routing information. Use this command to enter the hierarchical portion of your packet address. Do not enter your callsign or the first period of your packet address.

A hierarchical address consists of your state, country, and continent codes, separated by periods. Some areas (in large states particularly) use sub-state designators. For instance, a station in Rhode Island might use a hierarchical address of RI.USA.NOAM. This means Rhode Island (RI) which is in the United States (USA) which is in North America (NOAM). In this case, you should set the HTEXT to RI.USA.NOAM

If you are unsure of your hierarchical address, contact your local packet BBS sysop and ask what the proper addressing is for your location. Entering "%" will clear any text previously entered.

Note: Your PBBS will not forward or reverse forward unless HTEXT is set.

ID

immediate

When this command is entered an identification packet will be transmitted. This command may be used to insure that your station identification is the last transmission before taking the station off the air.

The ID packet is an unnumbered information <UI> packet whose data consists of your station identification as set in MYCALL and a listing of other modes, if you have activated them.

The MYCALL will be appended with "/R" if DIGIPEAT is ON, MYALIAS callsign appended with "/D", MYNODE callsign appended with "/N", and MYPBBS callsign appended with "/B". This packet will be addressed to "ID" and digipeated via any addresses specified in the UNPROTO command.

See also: hid, unproto

❖ **INTFACE {TERMINAL|NEWUSER|BBS|KISS|XKISS|HOST|GPS|MODEM}**

default NEWUSER

When set to NEWUSER, the TNC will operate with a standard terminal or computer running a terminal emulation program with a limited command set available. When set to TERMINAL, the full command set of the TNC is available.

When set to BBS, the TNC deletes certain messages (i.e. *** connect request, *** FRMR, etc) for greater compatibility with full-service BBS programs such as WØRLI, WA7MBL, CBBS, etc.

When INTFACE is set to KISS mode of operation, users interact with the TNC using software that implements the KISS code specified by Phil Karn (for more information, see the KISS Operation section of this manual).

When INTFACE is set to XKISS mode of operation, users interact with the G8BPQ extended KISS protocol (for more information, see the KISS Operation section of this manual).

When INTFACE is set to HOST, the TNC will talk to the attached computer using the Kantronics HOST mode and users communicate with the TNC using “host mode” software (for more information, see the HOST Operation section of this manual).

When INTFACE is set to GPS, the KPC-3 Plus will enter GPS mode upon power up. Data from the serial port will be parsed according to the GPSHEAD command, text will be placed into the LT buffers, and beacons will be transmitted according to the setting of the BLT commands.

When INTFACE is set to MODEM, the KPC-3 Plus mirrors at the RS232 port @ 1200 baud what it receives at the radio port (for more information, see the “modem mode” in the Modes of Operation chapter).

To have the KPC-3 Plus exit GPS Mode, connect a PC or terminal to it and issue three <Ctrl+C> characters (see COMMAND for details).

For instructions on exiting KISS, XKISS, HOST and MODEM modes, see the appropriate sections of the “modes of operation” chapter of this manual.

NOTE: After changing the setting of this command to KISS, XKISS, HOST, GPS, or MODEM a soft reset is required for the new mode to take effect (see RESET for details).

- **K**

immediate

This single letter command is synonymous with CONVERS. It is included as a single-keystroke convenience for entering Convers Mode.

See also: convers

KNTIMER n **(n = 0 - 255)**

default 15

If there is no activity (data) on a KA-Node circuit for n minutes, the KA-Node will disconnect both the input and output sides of the KA-Node circuit. Setting KNTIMER to 0 disables this feature.

See also: rnrtime

LCOK {ON | OFF}

default ON

When ON, no character translation occurs in the TNC. If OFF, lower case characters will be translated to upper case before being output to the terminal from the TNC. This case translation is disabled in Packet Transparent Mode.

LEDS {ON | OFF}

default ON

To conserve power, disable the indicators (LEDS) on the front-panel of the KPC-3 Plus by setting LEDS to OFF.

LFADD {ON | OFF}

default OFF

When ON, a line-feed will be appended to every carriage return received from the keyboard before being transmitted. When OFF, no line feeds will be added to the transmitted signal.

LFSUP {ON | OFF}

default OFF

When ON, this command suppresses any line-feed characters received from the other station, relying on your terminal program to properly advance to the next line. When OFF, the line-feeds received from the other station are not suppressed but are sent to your terminal as received.

See also: autolf, crsup

LGETCHAR $xx

Default $05 (CTRL-E)

LGETCHAR specifies a hot key functional in command or converse mode. This key causes the unformatted strings from the LTEXT buffers to be output to the RS232 port. This key performs the same function as the command “LT RESTART”. Set it to 0 to disable it.

LLIST [ON | OFF] [NONE | {+|-}callsign | callsign1,callsign2...]

default OFF

LLIST is used to determine which stations (callsigns) may NOT use your station for ANY purpose, including digipeating. When LLIST is ON, the TNC will NOT recognize those packets received with any callsign that appears in the LLIST’s (lid) list of callsigns. In addition, when LLIST is ON, you will not be able to connect to any station that is on that list.

The maximum number of callsigns allowed in LLIST is 10. Up to 10 callsigns may be entered at once, using the format shown after the command name, and the new list of from 1 to 10 callsigns will replace whatever list was there before. The LLIST command can be used to add a single callsign (+callsign), so long as there is room for the new callsign on the list (if there is not, the TNC responds with “EH?” and returns you to the command prompt). Similarly, you can remove a single callsign (-callsign) from the list. And to remove all items from the current list, enter LLIST NONE.

A callsign entered without a SSID will match any SSID of that callsign. To match a specific SSID only, enter the callsign with that SSID: for example, callsign-n, where n = 0-15.

LT {n [text] | RESTART} **(n = 1-4) (text up to 128 chars)**

default blank (for n = 1 - 4)

This command, used with GPS activity, fills the specified LOCATION Text (LT) buffer with “text”. n specifies which buffer to use (1-4). If GPSHEAD is set for this buffer, and if the INTFACE command is set to GPS, the buffer contents will be updated automatically by the GPS data.

To clear one of the 4 LT strings, enter “LT n%”, where “n” is the number of the LT string you want to clear.

When the command “LT RESTART” is executed, the LT buffers are output in an unformatted manner via the RS232 port, and the BLT timers are restarted. “Unformatted” means that only non-empty LT strings are output and that there are no preceding numbers indicating which strings are being output.

See also: blt, gpshead, ltp, ltrack

LTP n dest [via call1[,call2,...]] **(n =1-4)**

default GPS (for n = 1-4)

This command sets the destination callsign and the digipeaters used to transmit the LT strings. n specifies which string (1-4). The destination callsign defaults to GPS; however, you may want to change it to APRS, LOCATE, POSIT, or some other destination. Up to 8 digipeater callsigns may be specified.

See also: lt

LTRACK n [LT1] [LT2] [LT3] [LT4] [SYSOP] [TIME]

default 0

This command allocates memory for a special tracking buffer used to store LT messages. n defines the amount of memory (kbytes) set aside for the LT messages (limited by available RAM). Specifying the LT1 through LT4 parameters determines which messages will be stored in the tracking buffer. If SYSOP is specified, only a user with SYSOP privileges may access the tracking buffer, and if TIME is specified, a time stamp will be included, in front of the LT message.

Specified LTs are put in the LTRACK buffer based on the BLT settings.

The LTRACK buffer is accessed by connecting to the mailbox (MYPBBS) and using the mailbox's List Tracking buffer Command (LTR). To see the syntax of this command, type HELP at the BBS prompt.

To change LTRACK, you need to enter the entire command name, LTRACK, not just the short form (LTR).

See also: ltp, blt, GPS operation section

MALL {ON | OFF}

default ON

When ON, monitored packets include connected and unconnected packets between other stations. If OFF, only other station's unconnected packets (UI frames) will not be displayed.

This is a useful arrangement when stations are talking as a group in an unconnected configuration.

See also: monitor

MAXFRAME n **(n = 1 - 7)**

default 4

MAXFRAME sets an upper limit on the number of unacknowledged information packets, which can be outstanding at any one time. The TNC will send MAXFRAME number of packets in a single transmission, if they are available.

See also: paclen

MAXUSERS n **(n = 1 - 26)**

default 10

This command causes the TNC to allocate the memory required for the maximum number of simultaneous connections you wish to allow. Each connection uses a different stream. In order to direct what you want to say to a different stream you use the STREAMSW character. All streams may be used for outgoing packets, but USERS sets the number that may be used for incoming connections. Changing the value of MAXUSERS will cause the TNC to perform a “soft reset”. Note that you may not change the value of MAXUSERS while you are connected, since this would reset the TNC and cause all existing connections to be lost.

Note: Although you must spell out the entire command word (MAXUSERS) to change the current value of MAXUSERS, you can enter the short form, MAXU, if you just want to see the current value of MAXUSERS.

See also: status, streamsw, users

MBEACON {ON | OFF}

default ON

This command determines whether packets addressed to Beacon or ID will be displayed on the screen.

If you do not wish to display Beacon or ID packets, turn this command OFF.

See also: beacon, id, monitor

MCOM {ON | OFF}

default OFF

Supervisory (control) packets are not monitored unless MCOM is set ON and MON is ON. In addition if your station is connected, control packets are not monitored unless MCON is also ON. Control packets, when displayed, are distinguished from information packets by the “<>” and “<<>>” character sets appended to the packet header. With these characters, a letter is included to denote the type of control packet received, e.g. <<c>> - a connect request packet. The types of control packets are listed below.

<C> Connect request

<D> Disconnect request

<DM> Disconnected Mode

<UA> Unnumbered Acknowledge

In addition, the following bracketed information will be added to the Information packets as appropriate:

<UI> Unconnected Information frame

<Is> Information frame (connected);

s = send sequence number

See also: ax25l2v2, monitor, mresp

For details on control packets, see the book *AX.25 Amateur Packet-Radio Link-Layer Protocol Version 2.0 October 1984*, available from the ARRL.

MCON {ON | OFF}

default OFF

When OFF, and connected, the TNC will display only those packets addressed to you. Any header information displayed will be determined by the settings of STREAMEV and STREAMCA. When OFF, and not connected, all eligible packets (as determined by other monitor commands) will be displayed. When ON, all eligible packets will be displayed, whether connected or unconnected.

See also: monitor, streamca, streamev

❖ **MHEARD [SHORT | LONG | CLEAR]**

immediate

This command causes display of a list of stations heard. An asterisk, *, indicates that the station was heard through a digipeater. The date/time the station was last heard is also displayed.

If the S option is used, i.e. MHEARD S, then only the callsigns of the stations heard will be displayed.

If the L option is selected, all callsigns contained in the received packet including the digipeater paths, are displayed. For example:

NØGRG > ID 10/16/88 14:31:30

VIA TOP,KCI,WØXI

Here, your station heard NØGRG transmitting an ID packet. NØGRG was also using the digipeating path TOP, KCI, WØXI. If your station heard NØGRG via one of these other

stations, an asterisk would show by the call or alias of the last digipeater heard. When listed short, the asterisk would show beside NØGRG.

- Whenever PASSALL is ON, packets will not be checked for errors and no additions will be made to the MHEARD list.

If the C option is used, the list is cleared.

See also: daytime, passall

MHEADER {ON | OFF}

default ON

When ON, the headers are displayed for all monitored packets. When OFF, headers are not displayed and only data is output to the terminal. Since only I and UI frames have data, only these frames are displayed.

- To avoid confusion when MHEADER is OFF, use BUDLIST to restrict the monitored data.

See also: monitor, mcon, mbeacon, mall, suplist, budlist

❖ **MONITOR {ON | OFF}**

default ON

When ON, unconnected packets will be monitored unless prohibited by SUPLIST, BUDLIST, CONLIST, or LLIST. This will also allow monitoring of other packets if permitted by the other monitor commands. The MONITOR command acts as a master switch for the MALL, MCOM, MCON, MRESP, and MRPT commands. The addresses in the packet are displayed along with the data portion of the packet. Callsigns (to and from fields) are separated by a “>”; and the Secondary Station Identifier (SSID) is displayed if it is other than 0. If any data is contained in the monitored packet which does not follow the AX.25 protocol, it is displayed in curly braces on the header line. All monitor functions are disabled in the Transparent Mode.

When OFF, you will see just the data from those stations connected to you, no matter how other monitor commands are set. Any header information displayed will be determined by the settings of STREAMCA and STREAMEV.

See also: budlist, conlist, headerln, llist, mall, mbeacon, mcom, mcon, mresp, mrpt, mstamp, pid, streamca, streamev, suplist

MRESP {ON | OFF}

default OFF

When ON, monitored packets include the following AX.25 response packets, if MONITOR and MCOM are ON. If connected, MCON must also be on. The bracketed information will appear at the end of the header information. The “<” and “<<” characters are used to bracket and denote packets received as version 1 or version 2, respectively. For example, “<<RR1>>” denotes a version 2 packet. In addition upper case characters are used to designate commands (polls) and lower case characters are used to denote responses for RR, REJ, and RNR. For example, “<<rr1>>” is a response in version 2.

<FRMR> Frame Reject

<REJr> Reject, r = received sequence number

<RNRr> Device busy, r = received sequence number

<RRr> Receive Ready, r = received sequence number

In addition, the following bracketed information will be added to the Information packets as appropriate:

<Isr> Information frame (connected);

s = send sequence number,

r = received sequence number

See also: ax25l2v2, mcom, monitor

For more information, see the book *AX.25 Amateur Packet-Radio Link-Layer Protocol Version 2.0 October 1984*, which may be obtained from the ARRL.

MRPT {ON | OFF}

default ON

This command affects the way monitored packets are displayed. If ON, the entire digipeat list is displayed for monitored packets, and the station that relayed the packet is indicated with an asterisk. The MONITOR command must be ON for this command to work, and if connected MCON must also be ON. If OFF, only the originating station and the destination station callsigns are displayed for monitored packets.

See also: monitor

MSTAMP {ON | OFF}

default OFF

This command enables time stamping of monitored packets. The date and time information is then available for use for automatic logging of packet activity or other computer applications. The date and time are set initially by the DAYTIME command, and the date format is determined by the DAYSTR command. The MONITOR command must be ON for this command to work, and if connected MCON must also be ON.

See also: cstamp, daytime, monitor

MXMIT {ON | OFF}

default ON

When ON, the KPC-3 Plus will display transmitted packets as monitored data on your terminal. Repeated packets will be displayed as they are sent over the radio. The frames to be displayed will be controlled by the MONITOR, MCOM, MCON, and MRESP commands, and will obey the settings of TRACE, MSTAMP, HEADERLN, 8BITCONV and FILTER commands.

See also: monitor

MYALIAS xxxxxx-n **(n = 0-15)**

default (blank)

Setting this command to a callsign or character string enables digipeating by using the MYALIAS. Enter up to six characters (plus optional SSID) which are different than those used for MYCALL, MYNODE, MYPBBS, or MYREMOTE. For example: you may enter LAW-3 as the MYALIAS, which would be easier to remember for stations wishing to digipeat through a station in Lawrence, enroute to a station more distant. You may disable the alias digipeating with the command MYALIAS %.

See also: hid

❖ **MYCALL xxxxxx-n** **(n = 0-15)**

This command tells the TNC what its callsign is.

The extension n is called a Secondary Station Identifier (SSID) and is defaulted as 0, but may be any number from 0 to 15. All packets originated by the TNC will contain this callsign in the FROM address field. Any packets received by the TNC with this callsign in the TO address field or digipeat fields will be responded to appropriately (connect, disconnect, ack, digipeat, etc.).

- When the TNC is first turned on out of the box, or after a hard reset, it asks you for your callsign - there is no default. The callsign you enter then for MYCALL is automatically used to compute other IDs too (MYPBBS, MYNODE, and CWIDTEXT), adding an SSID for each. Changing MYCALL at the “cmd:” prompt does not affect MYPBBS, MYNODE, or CWIDTEXT

See also: cwidtext, digipeat, id, myalias, mynode, mypbbs, myremote

MYDGPS xxxxxx-n **(n = 0-15)**

default (blank)

This command accommodates differential GPS operation. In GPS mode only, if a UI packet is received addressed to the call in MYDGPS, the data from the UI packet is output on the RS232 port.

MYDROP n **(n = 0-15)**

default 0

This command sets the KISS address of the radio port in the KPC-3 Plus. A KISS frame with the upper nibble of the command byte set to this value will address this KPC-3 Plus.

See also: intface, KISS operation section

MYNODE xxxxxx-n **(n = 0-15)**

default mycall-7

Setting this command to a callsign or character string enables the KA-Node in the TNC. Enter up to six characters (plus optional SSID) which are different than those used for MYCALL, MYALIAS, MYPBBS, or MYREMOTE.

You must also have the NUMNODES command set to a non-zero value.

You may disable the KA-Node by setting MYNODE to the same as MYCALL, or setting NUMNODES 0.

See also: digipeat, ndwild, numnodes; KA-Node section

❖ **MYPBBS xxxxxx-n** **(n = 0-15)**

default mycall-1

This command is used to set the callsign (address) of your Personal Packet Mailbox. Entering a callsign with or without an SSID, -n, defines the address. The six characters and optional SSID must be different than those entered for MYCALL, MYALIAS, MYNODE, or MYREMOTE.

See also: pbbs

MYREMOTE xxxxxx-n **(n = 0-15)**

default (blank)

This command sets the callsign used for remote access of the TNC. Enter up to six characters (plus optional SSID) which are different than those used for MYCALL, MYALIAS, MYNODE, or MYPBBS. In addition, the RTEXT must be programmed with a

text string. A station that connects will be sent a series of numbers that must be decoded according to RTEXT.

See also: rtext; remote access section

NDHEARD [SHORT | LONG | CLEAR]

immediate

This command allows the operator to display a list of nodes whose ID packets have been heard by the TNC. The list includes KA-Nodes as well as K-Net, TheNet, NET/ROM and G8BPQ nodes.

K-Net, TheNet, NET/ROM and G8BPQ nodes are identified as:

ALIAS (CALLSIGN)

and Kantronics' KA-Nodes will be identified as:

MYNODE (MYCALL)

An asterisk "*" indicates that the station was heard through a digipeater. The date/time the station was last heard is also displayed. If the S option is used, i.e. NDHEARD S, then only the callsigns of the stations heard will be displayed. If the L option is selected, all callsigns contained in the received packet are displayed (see MHEARD for display format). If the C option is used, the list of nodes is cleared.

See also: mynode

NDWILD {ON | OFF}

default OFF

When OFF, the KA-Node will only recognize connect requests directed to the MYNODE call. When ON, connect requests to any SSID of the MYNODE call will be recognized as connects to the KA-Node, if that SSID is not being used for any other ID in the TNC.

See also: myalias, mycall, mynode, mypbbs, myremote

NETALIAS xxxxxx (up to 6 alpha-numeric characters)

default {blank}

The NETALIAS is an optional mnemonic that usually provides information about the location of the node. It exists only for the convenience of users. Network nodes automatically convert the NETALIAS to the associated NETCALL when setting up a circuit for a user. Changing or entering NETALIAS will not cause a soft reset.

NETBUFFS n **(n = 1 - 255)**

Default 32

The NETBUFFS command sets the number of node buffers available in the node. If the NETBUFFS reach zero, node operation may become unstable. Changing NETBUFFS will cause a soft reset as memory is reallocated. Each NETBUFF allocated uses 320 bytes of RAM.

NETCALL xxxxxx[-n] **(up to 6 alpha-numeric characters plus optional SSID)**

default {blank}

The NETCALL is the callsign of the K-Net node and can have an optional Secondary Station Identifier (SSID) extension of 1 to 15. The NETCALL must be entered before the K-Net can be operational and must be different from any other callsign used in the KPC-3 Plus.

Entering a % as the NETCALL will clear both the nodes and routes tables, but not change any node-related parameters. Entering or changing the NETCALL will cause a soft reset as memory is reallocated.

NETCIRCS n **(n = 1 - 64)**

default 5

This command limits the number of Network Circuits available in the node. A *circuit* is a connection between two nodes. The *circuit* will contain an *uplink* (originated by a user or server), a *crosslink* (the other network node), and usually a *downlink* (to another user or server). The *downlink* will not exist until requested by the originating *uplink* user. The *circuit* will exist until either the destination user or server or the originating user or server disconnects. The *circuit* will also be discontinued if one of the nodes determines that data can no longer be sent due to a retry time-out (L4N2), or if the inactivity timer (L4TI) expires. Changing the NETCIRCS will cause a soft reset as memory is reallocated. Each NETCIRC allocated uses 50 bytes of RAM.

NETDESTS n **(n = 1 - 255)**

default 25

Setting the NETDESTS determines the maximum number of destination nodes that can be stored in the K-Net node list. This list is sent to users when the NODES command is given to the K-Net node and also when the node does a nodes broadcast. Setting NETDESTS to a very large number (i.e. greater than 50 or so) is generally not a good idea since this can create long node broadcasts and long transmission times to users in response to a NODES command. Setting NETDESTS to a very low value can result with a desired node getting "bumped" out of the nodes table. After your K-Net has been in operation for a while, you should be able to determine how many of the destination

nodes that are listed in your nodes table are *reliable* paths. The value that you enter for NETDESTS should be greater than this number, allowing room for other nodes as they become available. In any case, never allow the node destination table to become so large that it is filled with unusable and unreliable destination nodes. Very large nodes tables (>50 or so) are simply baffling to new users, and most of the time unusable to even experienced "node hoppers".

Although setting NETDESTS will place a "hard limit" on the number of destination nodes that can appear in the nodes table, proper setting of the K-Net QUALITY and MINQUAL commands will allow the number of destination nodes listed in your nodes table to "float" with changing band conditions and assure that reliable destination nodes are still listed. Changing NETDESTS will cause a soft reset as memory is reallocated. Each NETDEST allocated requires 31 bytes of RAM.

NETLINKS n **(n = 1 - 64)**

default 10

The NETLINKS command sets the maximum number of *uplinks, downlinks,* and *crosslinks* available to the node. An *uplink* is a user that connects directly *to* the K-Net node without using another network node. A *downlink* is a connection *from* the K-Net node directly to a user. A *crosslink* is a connection between two nodes. Changing NETLINKS will cause a soft reset as memory is reallocated. Each NETLINK allocated requires 130 bytes of RAM.

NETROUTE n **(n = 1 - 32)**

default 5

This command sets the maximum number of neighbor nodes that can be listed with the ROUTES command. A neighbor node is a network node that your K-Net can hear "direct", without the benefit of being relayed by another network node. As with the NETDESTS parameter, if the setting of NETROUTE is too low, not all of your neighbors will be able to be listed. Setting NETROUTE to a higher number will allow you to see what the local node activity is and still have some room in reserve to handle itinerant nodes and nodes that will show up during a band opening. Changing NETROUTE will cause a soft reset as memory is reallocated. Each NETROUTE allocated requires 39 bytes of RAM.

NETUSERS n **(n = 0 - 26)**

default 5

NETUSERS sets the maximum number of *uplinks* and *downlinks* from the node. This command places a limit on how many users can access or be accessed by the node. The NETUSERS command will not cause a soft reset when changed or entered.

NEWMODE {ON | OFF}

default ON

When ON, the TNC will return to Command Mode if the station on the current I/O stream disconnects. The TNC will not return to Command Mode if the station disconnecting is on a different stream. When OFF, a disconnect will not cause the TNC to change modes.

See also: connect, disconnect, status

NOMODE {ON | OFF}

default OFF

When OFF and a connection takes place, the TNC will change to whatever mode is specified in CONMODE. When ON, the TNC stays in Command Mode after connecting to another station; that is, it does not immediately change to Convers or Transparent Mode.

See also: conmode, connect

NTEXT text **(up to 128 characters)**

default (blank)

This entry specifies customized text to be sent with the initial KA-Node sign-on message (when a remote station connects to a KA-Node). Enter any combination of characters and spaces up to a maximum length of 128. Entering a single '%' will clear NTEXT.

See also: mynode, numnodes

NUMNODES n **(n = 0-26, depending on available RAM)**

default 0

This command is used to set the number of allowable circuits through the KA-Node. For example, if you wish to allow up to 3 simultaneous circuits through the node, set NUMNODES 3.

- Changing the value of NUMNODES will cause a soft reset.

Approximately 4 kbytes of RAM is used for each circuit. The amount of RAM available will depend on how much RAM has been used for the PBBS, MAXUSERS, and MYREMOTE parameters. If you select n larger than available RAM will allow, a "Not enough RAM" message will be returned to you. Generally, set the amount of RAM required first for your PBBS (personal bulletin board) and then set the desired number of circuits. For example, if you have 32 kbytes of RAM and n is set to 5 you can only

have a 1kbyte mailbox. In this case, if n is larger than 5, a message saying “not enough RAM” will be returned to you.

Note: While you need to enter all the letters of the command name to change the value of NUMNODES, you can enter the short-form, NU, to see the current value of NUMNODES.

See also: mynode; KA-Node section

PACLEN n **(n = 0 - 255)**

default 128

This command specifies the maximum length of the data portion of a packet. The TNC will automatically send a packet when the number of input bytes reaches n. This value is used in both Convers and Transparent Modes. A value of 0 means 256 bytes.

See also: maxframe

PACTIME [EVERY | AFTER] n **(n = 0 - 255)**

default After 10

This command is used to pace the formation (and transmission) rate of packets in Transparent mode or in Convers mode if CPACTIME is ON. In addition, a PACTIME timer is necessary in transparent mode, since the SENDPAC character is ignored and treated as data. The timer is not started until at least one byte is entered for a packet; hence, a zero data length packet is never formed and sent.

When AFTER is specified, bytes are packaged (if any available) when input from the terminal stops for n*100 ms or when PACLEN is reached. When EVERY is specified, input bytes are packaged (if available) and queued for transmission every n*100 ms or when PACLEN is reached.

See also: cpactime, trans

PASS n **(n = $00 - $FF)**

default $16 <Ctrl+V>

This command selects the ASCII character used to allow the passage of any characters in Converse Mode.

For example, if you wish to send a COMMAND character (<Ctrl+C>) as part of the packet, you can do so by preceding it with the PASS character. The command character will be sent rather than returning the TNC to Command Mode. In Transparent Mode all characters are passed, there are no special functions except the one combination to get out of transparent mode. However, see TRFLOW.

PASSALL {ON | OFF}

default OFF

This command enables the attempt to display corrupted packets when PASSALL is ON. If a packet contains beginning and ending flags, the TNC will attempt to decode the address field as well as the data field and display the packet as specified by other commands such as MONITOR. MHEARD and NDHEARD logging are disabled when PASSALL is ON.

- When PASSALL is OFF, packets will be displayed only if the CRC (error checking) is correct and as specified by other commands such as MONITOR.

❖ **PBBS [n | STATUS]** **(maximum value of n depends on available RAM)**

default 5 (if RAM = 32 kbytes), 100 (128 kbytes), or 480 (512 kbytes)

Setting n greater than 0 allocates memory and activates the Personal Mailbox in the TNC. The amount of memory allocated will be n kilobytes, and may be limited by other functions that require memory (e.g. NUMNODES and MAXUSERS). Changing the size of the PBBS memory allocation will not affect the contents of the mailbox (messages will be preserved). If you attempt to set the PBBS smaller than is required for existing messages you will receive the message “Messages would be lost” and no change will be made to the mailbox size. In this case to reduce the size of the mailbox you must first either kill some of the messages or set PBBS 0 which will delete all messages; then set the new size. Using the PBBS n command with n equal to the current size will renumber the messages in the mailbox beginning with message number 1. If n is a different size, the messages will not be renumbered. This command causes a soft reset if n is different from its previous value.

Note: To change the value of PBBS you must enter the whole command name (PBBS), but to see the current value, you only need enter the short-form PB.

See also: cmsg, mypbbs, pbheader, pblo, pbperson, ptext; PBBS section

PBFORWRD [bbscall [VIA call1,call2,...call8]] [EVERY | AFTER n]

default NONE EVERY 0

This command will cause your PBBS to attempt to initiate a forward of any eligible mail to another BBS periodically. Any message in your PBBS which contains an @BBS field and is not being HELD (H) or has not previously been FORWARDED (F) is eligible for forwarding. If the keyword “EVERY” is used, the PBBS will attempt to forward once every n hours. If you specify the keyword “AFTER”, the PBBS will attempt to forward whenever a user disconnects from the PBBS, and every n hours after that. Setting the time interval will cause the PBBS to attempt to forward immediately.

After the PBBS has initiated a forward, it will also request a reverse forward if PBREVERS is ON. Note that full-service BBS systems must be configured to allow your station to reverse forward mail to you.

Note: Your PBBS will not forward or reverse forward unless HTEXT is set.

See also: pbrevers, htext

PBHEADER {ON | OFF}

default ON

When ON the routing headers received from a full service BBS will be stored in the PBBS mailbox. When OFF, these headers are not stored in the mailbox, allowing messages to require considerably less space. The routing headers are those lines you normally see in messages beginning with R:. Note that the PBBS will ignore all lines beginning with R: until it sees the first line that does not have R: in column one. From that point on, all of the message will be stored, even if a line begins with an R:.

See also: PBBS section

PBHOLD {ON | OFF}

default ON

When ON, any message received over the radio will automatically be held by your PBBS for you to review. You may then release the message for forwarding by editing it and changing the H flag (for example, to edit message number 4: E 4 H). When OFF, messages received over the radio are not held, but may immediately be forwarded from your PBBS. (Note that any message addressed TO or @ your MYCALL or MYPBBS call will be held regardless of the setting of PBHOLD.)

PBKILLFW {ON | OFF}

default OFF

When OFF, Private and Traffic messages will not be deleted (killed) from your PBBS after they have been forwarded, but will be marked with an F flag (forwarded). When ON, Private and Traffic messages will be deleted from the PBBS after they have been successfully forwarded.

PBLIST [ON | OFF] [NONE | {+|-}callsign | callsign1,callsign2...]

default OFF

The front panel of your TNC has a mail-status indicator that blinks when there is unread mail in your PBBS addressed to your callsign (MYCALL) or to your MYPBBS. And

several PBBS commands (LISTMINE, READMINE, and KILLMINE) operate on mail addressed to your callsign (MYCALL) or to your MYPBBS.

PBLIST allows you to add up to ten more callsigns to the set of callsigns included as “mine” for use with the “mail” indicator and the PBBS commands (LISTMINE, READMINE, and KILLMINE). The PBLIST command has two parameters: one for creating, changing, or deleting the list of callsigns on the PBLIST and the other for turning ON or OFF the use of the callsigns on the current PBLIST.

- SSID’s attached to callsigns are always ignored in the mailbox.

The maximum number of callsigns allowed in PBLIST is 10. Up to 10 callsigns may be entered at once, using the format shown after the command name, and the new list of from 1 to 10 callsigns will replace whatever list was there before. The PBLIST command can be used to add a single callsign (+callsign), so long as there is room for the new callsign on the list (if there is not, the TNC responds with “EH?” and returns you to the command prompt). Similarly, you can remove a single callsign (-callsign) from the list. And to remove all items from the current list, enter PBLIST NONE.

When you change PBLIST, the new list of callsigns takes effect immediately for use with the PBBS commands (LISTMINE, READMINE, and KILLMINE). But the new list will not be used with the mail-status indicator until after the next disconnect from the PBBS or the next reset.

When PBLIST is ON, the callsigns on the current PBLIST list are included as “mine” for use with the “mail” indicator and the PBBS commands (LISTMINE, READMINE, and KILLMINE). Your callsign (MYCALL) and your MYPBBS are also included as “mine” when PBLIST is ON.

When PBLIST is OFF, only your callsign (MYCALL) and your MYPBBS are included as “mine” for use with the “mail” indicator and the PBBS commands (LISTMINE, READMINE, and KILLMINE).

- Hint: Some uses of PBLIST to give you additional use of your “mail” indicator (and the PBBS commands shown above) would be to include on PBLIST:

  (1) callsigns of family members who are also hams,

  (2) old callsigns that people might still be using for you, and/or

  (3) callsigns that are common errors in trying to reach you (e.g., replacing a “zero” with an “0" in your callsign).

PBLO [OLD | NEW] [FIXED | VARIABLE]

default NEW VARIABLE

When set to OLD the PBBS will list messages to the user from oldest to newest (i.e. ascending numerical order). When set to NEW, the newest message will be listed first.

When the second parameter is set to FIXED, the user cannot change the listing order. When the second parameter is set to VARIABLE, the user may change the order in which messages will be listed by using the LO command within the PBBS.

See also: PBBS section

PBMAIL {ON | OFF}

default OFF

When ON, “YOU HAVE MAIL” will be sent to the terminal when new messages are available for MYCALL.

PBPERSON {ON | OFF}

default OFF

When OFF, the personal mailbox will allow messages to be sent to any callsign When ON, only messages addressed to the MYCALL , MYPBBS or PBLIST callsigns will be accepted over the radio, but a message entered from the terminal or by the SYSOP may be addressed to anyone.

See also: mycall, mypbbs, pbbs

PBREVERS {ON | OFF}

default ON

When ON, the KPC-3 Plus PBBS will request a reverse forward from another BBS after it has finished forwarding any mail to the other BBS. This does not affect the ability of another BBS to reverse forward from your PBBS, but only affects the action taken by the KPC-3 Plus after it has initiated a forward (PBFORWRD command). Note: Your PBBS will not forward or reverse forward unless HTEXT is set.

See also: pbforwrd

PBSIZE n **(n = 0 - 65535)**

default 0

This command controls the maximum size of new messages entered into the PBBS. A value of zero means there is no maximum.

PBUsers n **(n = 1 - 10)**

default 1

This command controls the maximum number of connects allowed to the PBBS. This command causes a soft reset if n is different from its previous value.

PERSIST n **(n = 0 - 255)**

default 63

This command, used with SLOTTIME, implements an algorithm for channel access to send packets. The algorithm used to determine whether or not to transmit using the PERSIST/SLOTTIME method has been shown to be considerably more efficient than the DWAIT method used by most standard AX.25 packet stations. The result of using the persistence algorithm is increased throughput under most channel conditions.

The value of “n” is used to determine if a packet will be sent after SLOTTIME expires. For example, let’s assume a PERSIST setting of 63 and a SLOTTIME setting of 10. This slottime setting corresponds to 100 milliseconds. When the TNC detects that the channel is clear and available (no carrier is detected), it starts a timer (SLOTTIME). When the timer expires, (100 ms in our case), the TNC generates a random number between 0 and 255. If the generated number is equal to or less than the PERSIST value, the TNC keys up the transmitter and sends the data packet. With our setting of 63 the odds of this occurring after the first slottime are 1 in 4. (Actually the probability is PERSIST plus 1 divided by 256.) If the TNC generated random number is greater than PERSIST, the TNC restarts the timer and waits for the timer to expire again before generating a new random number. This is repeated until the TNC gains channel access and sends its packet of information.

Making SLOTTIME smaller will cause the TNC to generate the random number more frequently, whereas raising the PERSIST value will give a better chance (improve the odds) of transmitting the data. Through careful choice of these values, it is possible to improve data throughput while at the same time permitting shared channel usage by other packet stations. The persistence algorithm has been added on top of the DWAIT algorithm.

See also: slottime

PHEARD [CLEAR]

immediate

This command will display a list of the last 10 stations that have connected to your KPC-3 Plus PBBS. The list will show the callsign of the connecting station and the last time they connected and disconnected.

To clear the PHEARD list of all current entries, enter PHEARD CLEAR.

See also: passall

PID {ON | OFF}

default OFF

When OFF, only those packets with a protocol ID of $F0 (pure AX.25) are displayed. When ON, all packets are displayed. Some of the information in non-AX.25 packets (for example: TCP/IP, NET/ROM or TheNet) can cause some computers to lock up. Net/Rom, TheNet and G8BPQ nodes have a PID of $CF, TCP/IP uses $CC and $CD, and standard AX.25 is $F0.

PMODE {CMD | CONV | TRANS}

default CMD

The PMODE command controls the mode your KPC-3 Plus will be in when it is first powered up or reset. When set to CMD, the KPC-3 Plus will produce a sign-on message followed by the command prompt. When set to CONV, the unit will be in the Convers Mode. Setting PMODE TRANS sets the KPC-3 Plus in Transparent Mode. Exiting Transparent Mode with three <Ctrl+C>s will put the KPC-3 Plus in Command Mode.

PTEXT text (up to 128 characters)

default (blank)

This entry specifies the customized text sent with the initial PBBS (personal mailbox) sign-on message. Enter any combination of characters and spaces up to a maximum length of 128. Entering a single “%” will clear PTEXT. Don’t use the “>” character in your PTEXT, as this is reserved by BBS systems for their prompt.

See also: pbbs

RANGE a-b/c-d/e-f/g-h/i-j

default 0-255/0-255/0-255/0-255/0-255

The 5 A/D inputs are normally read by the microprocessor as a single byte value from 0 to 255, representing a voltage from 0 to +5 V on the pin. You can use the RANGE command to mathematically convert that single byte value to any offset and scale you want. If you would like the input to be expressed as a voltage with 2 decimal precision, for example, set the range for that analog pin as 0-5.00. If you connect a temperature sensor to an A/D pin which outputs 0-5 V for temperatures of –40 to 100, set the range for that input to –40 – 100.0. A pressure sensor that gives 0-4 V for 1 PSI would use a range of 0-1.25.

See also: analog, telemetry

REDISPLA n **(n = $00 - $FF)**

default $12 <Ctrl+R>

This command is used to change the REDISPLAY input editing character. The parameter n is the ASCII code for the character you want to type in order to REDISPLAY the packet or command currently being entered.

You can type this character to cause the TNC to redisplay the packet you have begun. When you type the REDISPLAY-packet character, the following things happen: First, type-in flow control is released (if FLOW was enabled). This displays any incoming packets that are pending. Then a \ (backslash) character is displayed, and the packet or command you have begun is redisplayed on the next line. If you have deleted and retyped any character, only the final form of the packet will be shown. You are now ready to continue typing. Incoming packets will continue to be displayed until you type the next character of your current message.

You can use the REDISPLAY-packet character to see a “clean” copy of your input if you are using a printing terminal (or have BKONDEL OFF) and you have deleted characters. The REDISPLAYed packet will show the corrected text.

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

See also: bkondel, canline, canpac, flow

RELINK {ON | OFF}

default OFF

When OFF, the TNC operating with AX25L2V2 ON does not attempt to automatically reconnect. When ON, the TNC operating with AX25L2V2 ON will attempt to automatically reconnect after RETRY is exceeded.

The KA-Node and the PBBS will never attempt to reconnect regardless of the setting of this command. If using AX.25 Level 2 Version 1 (AX25L2V2 OFF) this command has no effect.

See also: ax25l2v2, retry, tries

❖ **RESET**

immediate

This command is used to perform a soft reset. Contents of the mailbox (PBBS) are kept, and the NDHEARD and MHEARD logs are not cleared. Any existing connections, which are not conpermed, will be lost, even though the other station still indicates a connection to your station. The initial sign-on message will be displayed.

See also: intface, maxusers, myremote, numnodes, pbbs, restore

RESTORE DEFAULT

immediate

When RESTORE D is given, the KPC-3 Plus will revert to its factory default settings, perform its AUTOBAUD routine, and erase PBBS memory (deleting all messages and non-default PBBS parameter settings).

See also: reset

RETRY n **(n = 0 - 15)**

default 10

This command specifies the number of packet retries. Packets are re-transmitted n times before an operation is aborted. The time between retries (when no acknowledgment is received) is specified by the command FRACK.

See also: ax25l2v2, frack, relink, tries

RING {ON | OFF}

default ON

When ON, three bell characters ($07) are sent to the terminal with each "*** CONNECTED TO" message (when another station initiates the connect).

RNRTIME n **(n = 0 - 255)**

default 0

RNRTIME is set in 10-second increments. If a connection stays in a remote device busy state (continues to receive RNR frames) for RNRTIME, the TNC will disconnect. If a KA-Node connection stays in a remote device busy state for RNRTIME the KA-Node will disconnect the input and output sides of the KA-Node circuit. Setting RNRTIME to 0 disables this function.

See also: mresp

RPRINT text **(text up to about 250 characters)**

immediate

This is an immediate command which sends the "text" string to the device attached to the RS232 port (e.g. a GPS unit). It is intended to allow a remote SYSOP (connected to

the KPC-3 Plus with the MYREMOTE callsign) to send a string to a unit attached to the KPC-3 Plus serial port. The string content could be used to change the configuration of a GPS unit.

RTEXT text **(text up to about 250 characters)**

default (blank)

This command sets the password string for use when accessing the MYREMOTE or when performing SYSOP functions in the PBBS remotely. When you connect to the MYREMOTE or attempt to enter the SYSOP mode of the PBBS, the TNC will send three sets of numbers, one of which must be properly decoded (matching position number with string character) using this string.

For instance, if the RTEXT is set to “This is my RTEXT string” and you connect to the MYREMOTE, the TNC would send three sets of random numbers. The numbers would look like:

5 20 14 7 18 3

7 1 4 14 8 19

9 3 8 12 22 1

You must then pick ONE of these lines and decode the password string (i.e., assign letters to each letter position). Let’s say you choose to decode line 3 (9 3 8 12 22 1). Rewriting the RTEXT string to make this easier we have:

```
          11111111112222
123456789012345678901 23
This is my RTEXT string
```

Now, decoding the string, character 9 is “m” character 3 is “i”, character 8 is “ ” (i.e., blank), character 12 is “R”, character 22 is “n”, and character 1 is “T”. You must send the following in response to the string of numbers:

mi RnT

If correct, you’ll receive the MYREMOTE prompt (“prompt:”).

- Note that case is significant and spaces are considered valid characters.

If you fail to properly decode the password, the remote KPC-3 Plus will send three new lines of numbers. You will be given a maximum of three attempts to properly decode the password string. If you fail in three attempts, the TNC will disconnect. After failing in three attempts, the MYREMOTE is disabled for 15 minutes.

SCREENL n **(n = 0 - 255)**

default 0

This value is used to format what is sent to your terminal. A “CR” sequence is sent to the terminal at the end of a line when n characters have been printed. A value of zero inhibits this action.

See also: autolf

SENDPAC n **(n = $00 - $FF)**

default $0D <Ctrl+M> (CR or ENTER)

This command specifies a character that will force a packet to be sent in Convers Mode. In the Convers Mode, packets are sent when the SENDPAC character is entered or when PACLEN is achieved.

See also: cpactime, cr

SLOTTIME n **(n = 0 - 255)**

default 10

n specifies the amount of time, in 10 millisecond increments, between successive tries of the persistence algorithm.

See also: persist

START n **(n = $00 - $FF)**

default $11 <Ctrl+Q>

This command specifies the character sent by the computer to the TNC to restart output from the TNC. If set to $00 only hardware flow control will be used. For software flow control, set this parameter to the character the computer will send to restart data flow.

See also: stop, xflow, xoff, xon

❖ **STATUS [LONG]**

immediate

Entry of the STATUS command will display the number of bytes available (i.e., free bytes) in the RS232 input buffer, the current I/O stream, and any streams having a connected status.

Entering LONG will cause the TNC to display the status of all streams allowed by MAXUSERS, PBBS, NUMNODES, and MYREMOTE.

The current KA-NODE input and output (IO) stream is also indicated. A pound sign '#' (octothorpe) indicates that there is unacknowledged data in the buffers for that stream. The number immediately following the # is the number of bytes outstanding. The number in parentheses is the number of packets not yet acknowledged.

The following is an example of a display (using STATUS LONG). A, B, C, through J (i.e., MAXUSERS = 10) indicate the stream. "A" stream is connected to KEØSM and has 50 bytes in 2 packets that are unacknowledged. The current stream (IO) is the "B" stream, which is connected to NØKN. KBØNYK is connected on circuit "A" of the KA-Node. All other streams, node circuits, the BBS, and the remote access are disconnected. Appending "/P" to the callsign indicates that the connection is permanent (see CONPERM).

```
cmd: status long
FREE BYTES 1661
A stream - #50(2) CONNECTED to KEØSM
B stream - IO CONNECTED to NØKN/P
C stream - DISCONNECTED
J stream - DISCONNECTED
BBS DISCONNECTED
REM DISCONNECTED
Ain CONNECTED to KBØNYK
Aout DISCONNECTED
Bin DISCONNECTED
Bout DISCONNECTED
```

See also: conperm, maxusers, myremote, numnodes, pbbs, streamsw

STOP n **(n = $00 - $FF)**

default $13 <Ctrl+S>

This command specifies the character sent by the computer to the TNC to stop output from the TNC. If set to $00 only hardware flow control will be used. For software flow control set this parameter to the character the computer will send to stop data flow.

See also: start, xflow, xoff, xon

STREAMCA {ON | OFF}

default OFF

When receiving packets addressed only to you (MONITOR OFF and/or MCON OFF), setting this command ON will enable the display of the callsign of the connected-to

station following the stream identifier of the connection (controlled by STREAMEV). This is especially useful when operating with multiple connections allowed.

See also: mcon, monitor, streamev

STREAMEV {ON | OFF}

default OFF

When OFF, the stream indicator is displayed only when a change in streams occurs. When ON, the stream indicator will be displayed with every incoming packet. This command takes effect only when receiving those packets addressed to you and MCON is OFF.

See also: mcon, monitor, streamca, streamsw

STREAMSW n **(n = $00 - $FF)**

default $7C (|)

This command selects the character to be used to signify that a new “stream” or connection channel is being addressed. To change streams you must type this streamswitch character followed immediately by the stream designator. The stream designator is an alphabetic character A through Z, or a through z, limited by the value of MAXUSERS.

- If STREAMSW is set to the dollar sign ($24) you will need to enter parameter values in decimal. Alternatively, precede the $ with the PASS character in order to enter hex numbers.

The character selected can be PASSed in the Convers Mode by using a special PASS character, and will always be passed as data in the Transparent Mode. If operating in the Transparent Mode and you wish to change streams, you must first return to the Command Mode.

See also: maxusers, pass, status

SUPLIST [ON|OFF] [NONE | {+|-}call | call1,call2,..]

where call={callsign | {<|>}callsign |

callsign{>|<>}callsign}

default OFF NONE

SUPLIST is used to determine which received packets will not be displayed, i.e. suppressed. When OFF or NONE, SUPLIST will not prevent the display of packets, even if it has callsigns in its list.

The maximum number of callsigns allowed in SUPLIST is 10. Up to 10 callsigns may be entered at once, using the format shown after the command name, and the new list of from 1 to 10 callsigns will replace whatever list was there before. The SUPLIST command can be used to add a single callsign (+callsign), so long as there is room for the new callsign on the list (if there is not, the TNC responds with “EH?” and returns you to the command prompt). Similarly, you can remove a single callsign (-callsign) from the list. And to remove all items from the current list, enter SUPLIST NONE.

A callsign entered without a SSID will match any SSID of that callsign. To match a specific SSID only, enter the callsign with that SSID: for example, callsign-n, where n = 0-15.

Selective display is enabled when one or two callsigns are combined with a “>” and/or “<” character. For example, entering “>callsign” will disable display of packets TO that callsign (assuming SUPLIST is ON), entering “<callsign” will disable display of packets from the callsign, entering “callsign1>callsign2" will block display of packets from callsign1 to callsign2, and entering ”callsign1<>callsign2" will block display of packets from callsign1 to callsign2 and vice versa.

- Note that entering call>call or call<>call counts as two of the 10 maximum allowed callsigns.

See also: monitor

SWP u,d,t

default 17,17,108

This command sets the parameters used for software carrier detect. The first number (u) is used to increment a counter when a valid mark/space or space/mark transition occurs in the received signal (i.e. transition occurs at the beginning of a bit time). The second number (d) is a penalty subtracted from the counter when a transition occurs in the middle of a bit time. The t value is the threshold value - when the counter total reaches this value, the carrier detect will be set true.

The values need to be separated by commas as they are entered (as shown). Also as shown, spaces between values are not necessary—but entering spaces after the commas is OK.

Once carrier detect is active, the counter must drop to 0 before carrier detect is again made false.

See also: cd

TELEMETRY n **(n = 0 - 255)**

default 0

A telemetry beacon is sent every n*10 seconds. If n = 0, the beacon is disabled. The telemetry beacon is a UI frame sent to BEACON. If there is a current UNPROTO digipeat path, it will be used to transmit the telemetry beacon.

The data in the UI frame is in the form: “T#nnn,an0,an1,an2,an3,an5,bbbbbbbb”, where:

- “nnn” is a decimal count for the telemetry beacon and is incremented each time the beacon is sent. When first used, or after the unit is reset, the beacon counter begins at 001 (T#001). After reaching 999 (T#999), the counter cycles around to 000 (T#000).

- “an0", ”an1", “an2", ”an3", and “an5" are the decimal readings on the corresponding analog inputs of the 68HC11F1 (the KPC-3 Plus microprocessor).

+ Note: APRS expects leading zeroes for the analog readings; you must set RANGE to 0-255 to use the APRS format.

- “bbbbbbbb” is the binary reading of the 8 pins on PORT E of the 68HC11F1 (PORT E is a shared port with the analog inputs), where “b” = “1" means a Logic High on the pin and ”b" = “0" means a Logic Low on the pin. The first ”b" in the string of 8 binary digits is the most significant bit of PORT E and the last “b” is the least significant bit of PORT E.

+ Hint: For details on the 8 pins on PORT E, see the KPC-3 Plus schematic.

See the documentation on APRS for more information on the use of telemetry along with GPS.

See also: analog, range

TRACE {ON | OFF}

default OFF

When ON, all received frames are displayed in their entirety, in hexadecimal, including all header information. All packets, which are eligible for monitoring also, will be displayed in normal text.

TRANS

immediate

This command causes immediate exit from Command Mode into Transparent Mode. The current link state is not affected. There are no special editing characters, all characters are sent out as received. To get out of Transparent, send the TNC a modem break if BREAK is ON, or see CMDTIME for a special keyboard sequence.

See also: break, cmdtime, command

TRFLOW {ON | OFF}

default OFF

This command allows the TNC to respond to software flow control from the computer while in the Transparent Mode. When TRFLOW is OFF, software flow control is not used in the Transparent Mode. Hardware flow control will be expected from the computer by the TNC. The computer program needs to use hardware flow control, and the RS232 cable needs to be wired with CTS and RTS connected. When ON, software flow control is enabled and the START and STOP characters are sent by the computer to the TNC to control the flow of data. When START and STOP are set to $00, hardware flow control must be used. If not zero, the TNC will respond to the computer's START and STOP characters, and remain transparent to other characters from the terminal or computer for flow control.

When START and STOP are set for software flow control (normally <Ctrl+Q> and <Ctrl+S>) all characters can be received in Transparent Mode (including the START and STOP characters) by setting TRFLOW ON and TXFLOW OFF. You will not, however, be able to send the START and STOP characters, since the TNC will interpret them as flow control.

See also: trans, txflow, xflow

TRIES [n] **(n = 0 - 15)**

The TRIES command will display and optionally set the number of attempts, which have been made to re-send a packet (on the current stream) which failed to reach its destination. For instance, if RETRY is set to 10, TRIES will show how many attempts have already been made to pass the data. For example, if TRIES were to show 8, "TRIES 3" would reset the counter to make the TNC believe that it had only tried 3 times so far, thus allowing 7 more attempts before the RETRY limit is exceeded.

See also: retry

❖ **TXDELAY n** **(n = 0 - 255)**

default 30

This command sets the transmitter key-up delay as 10*n ms. This setting establishes the time delay between the application of push-to-talk and the start of packet data to the transmitter. Flags (characters to begin packet) are sent during the delay. This command needs to be set long enough to give your transmitter time to come to full power before data is sent. If set too short the beginning of the packet will be chopped off and another station will never be able to decode you. If set too long, additional flags at the beginning (heard as a repetitive sound) waste air time. It may be necessary to increase your TXDELAY to allow the receiving station sufficient time to switch from transmit back to receive to detect your signal.

TXFLOW {ON | OFF}

default OFF

This command allows the TNC to send software flow control (XON and XOFF) to stop and restart the flow of data from the computer while in the Transparent Mode. When TXFLOW is OFF, hardware flow control must be used between the computer and TNC. RTS and CTS must be connected between the TNC and computer for hardware flow control. When TXFLOW is ON, software flow control between the TNC and computer in Transparent Mode will depend on the setting of XFLOW. XFLOW ON enables software flow control, XFLOW OFF disables it. When software flow control is enabled, the TNC will send the XON and XOFF characters to the computer to control data flow.

When set for software flow control, all characters can be sent in Transparent Mode (including the XON and XOFF characters) by setting TXFLOW ON, XFLOW ON and TRFLOW OFF. You will not, however, be able to receive the START and STOP characters, since your terminal program should interpret them as flow control.

See also: trans, trflow, xflow

UIDIGI ON [+|-] call1[,call2[,call3[,call4]]]

default OFF NONE

Up to 4 call signs can be specified for special digipeater duty. If any of the UIDIGI calls appears in the to-be-digipeated field of a UI packet, and if MYCALL does not appear in the source field or any of the has-been-digipeated fields, the UIDIGI call in the to-be-digipeated field will be replaced by MYCALL with the H bit set and the packet will be digipeated.

See also: dwait, persist, uidwait, unproto

UIDUPE n **(n = 0–255)**

default 0 (disabled)

Timer for UI duplicate checking. Use of this “do not duplicate within n seconds” feature helps control the number of UI frames digipeated.

See also: uiflood and uitrace.

UIDWAIT [ON|OFF]

default OFF

When UIDWAIT is OFF, “special” digipeat packets (those formed by UIDIGI, UIFLOOD, or UITRACE only) have their usual channel access; there is no wait DWAIT or slottime added before transmission once the channel is clear. However, if UIDWAIT is set ON,

the packets waiting to be digipeated will be subject to the same wait times as not-to-be-digipeated packets awaiting transmission.

- By subjecting “special” to-be-digipeated packets to a delay determined by slottime and persist, it is more likely that to-be-digipeated packets of two or more stations in the same vicinity would not collide. This may be good if one wants to guarantee that a digipeated packet will “make it out” of its neighborhood but bad if one wishes to limit the number of times a packet is redigipeated, such as in APRS applications.

See also: dwait, persist, uidwait

UIFLOOD name[,time[,{FIRST | ID | NOID}]] **(name = 5 char max)(time = 0-255)**

default disabled,30,NOID

When a UI frame is received with a call in the to-be-digipeated field of the form ‘name’ x-y where x is a number (1-7) appended to ‘name’ and y is an ssid (1-7), the ssid is decremented and the UI frame is digipeated without setting the H bit. When the packet is digipeated, a checksum is formed over the source, destination, and data fields of the packet. This checksum is kept for time seconds (0-255). If an incoming UI packet is eligible for digipeating as above, but its checksum matches one of those being saved, the packet is discarded (not digipeated). The buffer holds a maximum of 64 checksums. If the optional parameter ID is selected, the MYCALL call sign is inserted in an additional digipeater address field with its H bit set.

See also: uidwait

UITRACE name, n **(name = 5 char max)(n = 0-255)**

default disabled, 30

When a UI frame is received with a call in the to-be-digipeated field of the form ‘name’ x-y where x is a number (1-7) appended to ‘name’ and y is an ssid (1-7), and MYCALL does not appear in the source field or any of the has-been-digipeated fields, MYCALL with the H bit set is inserted before the to-be-digipeated field, the ssid of the to-be-digipeated field is decremented, and the UI frame is digipeated without setting the H bit of the to-be-digipeated field. The value of n sets a delay time between duplicate digipeats. Use of this “do not duplicate within n seconds” feature helps control the number of UI frames digipeated.

- If the packet should already have 8 digipeater fields, MYCALL is not inserted.

See also: dwait, persist, uidwait

UNPROTO {call1 [VIA call2,call3..call9] | NONE}

default CQ

call1 = destination address (this is really just a “dummy” address, as no connection takes place, people often put their name or CQ here)

call2 ... call9 = optional stations to be digipeated through. A maximum of 8 digipeat addresses (callsigns or aliases) can be specified. This is referred to as a path.

Each callsign may also have an optional Secondary Station Identifier (SSID) specified as -n, where n = 1 - 15. The digipeat callsigns are specified in the order in which they are to relay transmitted packets. This command is used to set the digipeat and destination address fields for packets sent in the unconnected (unprotocol) mode. Unproto packets do not receive an acknowledgment and are not retried. They are sent as Unnumbered I-frames <UI>. The digipeater list is also used for BEACON and ID packets. If UNPROTO is “NONE”, no unconnected packets will be sent except for BEACON and ID. Unconnected packets sent from other units can be monitored by setting MONITOR ON. If you are connected, you must also set MCON ON.

See also: beacon, id, monitor, mrpt, xmitok

USERS n (n = 0 - 26)

default 1

This command specifies the channels (streams) which may be available to incoming connect requests. For example, if USERS = 5 then an incoming connect request will connect to the lowest channel A - E, if any of these channels are in the unconnected state. If none of the 5 channels are available (all of them are connected), a <DM> packet will be sent back to the requesting station and the message “*** connect request: (call)” will be output to your terminal, if permitted. If USERS is set to 0 no one will be able to connect to you. If USERS is set higher than MAXUSERS, the extra is ignored and the message “USERS LIMITED BY MAXUSERS” will be displayed.

See also: intface, maxusers, streamsw

❖ **VERSION**

immediate

This command causes the TNC to display its current version number along with the name of the unit.

WEFAX n

immediate

This command permits the reception of Weather Facsimile. The audio input is sampled n times per second, and a black/white decision is made on each sample. Each sample becomes one bit of the data sent over the serial port to the computer. A special computer program must be used to display the WEFAX pictures.

See also: daytweak, WEFAX Reception in Modes of Operations section

XFLOW {ON | OFF}

default ON

When ON software flow control will be implemented according to the settings of START, STOP, XON, XOFF. For normal software flow control set XFLOW ON, START $11, STOP $13, XON $11, XOFF $13. The TNC expects the computer or terminal to respond to the flow control characters XON and XOFF, and the TNC responds to the START and STOP characters from the computer. When XFLOW OFF, the TNC will only use and recognize hardware flow control lines (CTS and RTS) to start or stop data. The RS232 cable must be wired appropriately. If the software flow control characters are set to $00, software flow control is not possible.

In the Transparent Mode flow control is also determined by the settings of TRFLOW and TXFLOW.

See also: start, stop, trflow, txflow, xoff, xon

XKCHKSUM {ON | OFF}

default OFF

This command only applies when INTFACE is set to XKISS. The value of XKCHKSUM controls whether or not the XKISS CHECKSUM mode is active.

See also: xkpolled

XKPOLLED {ON | OFF}

default OFF

This command only applies when INTFACE is set to XKISS. XKPOLLED controls whether or not the XKISS POLLED mode is active.

See also: xkchksum

XMITLVL n **(n = 0-502)**

default 100

This command may be used to set the modem drive level. You may find it more convenient to use the CAL command in that it allows you to continuously adjust drive level by holding down the + or - key while in calibrate mode.

The voltage range set by the XMITLVL command is from 1 mV to 4 V p-p. From a count of 0 to 255, the voltage is increased from 1 mV to (approximately) 130 mV, or 0.50 mV

per step. From a count of 256 and above, the voltage is increased linearly but in larger steps (approximately 15 mV per step) until 4 V (max) is reached at a count of 502.

XMITOK {ON | OFF}

default ON

When ON, transmitting functions are enabled. If turned OFF, transmitting is inhibited while all other functions of the TNC are unchanged.

XOFF n **(n = $00 - $FF)**

default $13 <Ctrl+S>

This command selects the character sent by the TNC to the computer to stop input from the computer. If set to $00, hardware flow control must be used. For software flow control set this parameter to the character the computer expects to see to stop sending data to the TNC.

See also: xflow, xon

XON n **(n = $00 - $FF)**

default $11 <Ctrl+Q>

This command selects the character sent by the TNC to the computer to restart input from that device. If set to $00, hardware flow control must be used. For software flow control set this parameter to the character the computer expects to see to restart sending data to the TNC.

See also: xflow, xoff

Appendix A: Advanced Installation

This appendix gives you additional information on installation and configuration beyond that given in the chapter on **"Installing your KPC-3 Plus".** We assume you have already read that chapter.

Precautions

The KPC-3 Plus is grounded through its connections to your transceiver, computer, and power supply. Make sure your transceiver is properly grounded and your computer has equal ground potential. Follow the grounding instructions in your transceiver manual.

The multi-conductor cable provided with your KPC-3 Plus is shielded. If you use other cabling, be certain it is shielded. We do not recommend the use of unshielded RS232 ribbon cable in an amateur radio environment.

Lithium batteries can explode or leak if heated, disassembled, recharged, exposed to fire or high temperature, or inserted incorrectly.

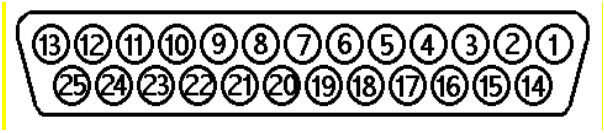
Cable Wiring

Caution: Make sure the power to the transceiver, computer, and KPC-3 Plus is OFF before connecting any cables.

The KPC-3 Plus uses a standard RS232 serial port with a DSUB-25 connector configured as DCE equipment.

Diagram of Pin Numbers on DSUB-25 Connector

The following diagram shows the pin numbering on the DSUB-25 connector looking at the back of the KPC-3 Plus:



Connecting to the Computer (DSUB-25)

The KPC-3 Plus communicates with the computer using RS232 signaling. This is compatible with PC compatible computers, Macintosh, dumb terminal, and most other terminal devices. If your computer or terminal uses TTL levels (such as the Commodore 64), you will need an after-market converter to supply the RS232 signals required by the KPC-3 Plus.

Cable Wiring

All Kantronics software requires five wires between the computer and the KPC-3 Plus. The pins that must be connected from your KPC-3 Plus to the computer for these programs are TXD, RXD, SG, RTS, and CTS. If you are using other software, you should refer to that documentation to determine the cable requirements for proper operation. Some software does not support hardware flow control and may fail to operate if the RTS and CTS lines are connected.

The KPC-3 Plus serial port connector has 25 pins, which are configured at the factory as:

Pin Name	KPC-3 Plus Pin Number	Computer Pin Number (DSUB-25)	Computer Pin Number (DSUB-9)
FG	1	1	N/A
TXD	2	2	3
RXD	3	3	2
RTS	4	4	7
CTS	5	5	8
DSR	6	6	6
SG	7	7	5
DCD	8	8	1
DTR	20	20	4

NOTE: FG (Frame Ground) and SG (Signal Ground) are tied together in the KPC-3 Plus.

The purpose of the pins (by name) is:

FG - FRAME GROUND: This pin is attached to the chassis of the equipment as a safety ground.

SG – SIGNAL GROUND: Common line for signals.

TXD - TRANSMIT DATA: This line carries the data from your computer to the KPC-3 Plus.

RXD - RECEIVE DATA: This line carries the data from the KPC-3 Plus to your computer.

RTS - REQUEST TO SEND: The KPC-3 Plus checks this line to see if it is permitted to send data to your computer. This pin is controlled by your computer software program to provide this information to the KPC-3 Plus. This pin is used for hardware flow control.

CTS - CLEAR TO SEND: The KPC-3 Plus uses this line to signal your computer when it can no longer accept data from the computer, or to signal that it is again ready to accept data. This pin is used for hardware flow control.

DSR - DATA SET READY: This line is set high to indicate that the modem is ON.

DCD - DATA CARRIER DETECT: The KPC-3 Plus uses this pin to signal the status of the current I/O stream to your computer. If you are connected to another packet station on the current I/O stream, this line will have a positive voltage on it. If you are disconnected, the voltage on this line will be negative.

DTR - DATA TERMINAL READY: Although interconnected via a buffer IC to the processor, this line is currently ignored.

Optional Wiring

It is possible to re-configure the DSUB-25 connector to allow some additional functions through this connector.

Applying Power through the DSUB-25 connector.

If desired, you may apply dc power to the KPC-3 Plus from Pin 13 of the DSUB-25 connector. To enable this, connect J6 center pin and pin 2, J7 center pin and pin 1.

Resetting the KPC-3 Plus through the DSUB-25 connector.

If desired, EXT-IN via pin 13 of the DSUB-25 connector may be configured to allow you to perform a soft reset. This is accomplished by connecting J6 center pin and pin 1, J7 center pin and pin 1. A momentary contact to ground will cause a soft reset.

Hardware handshaking with DSR and DTR.

It is possible that some terminals or computers use the DTR and DSR pins for hardware handshaking instead of the RTS and CTS pins. When this is the case, you must connect the DTR terminal output to RTS on the KPC-3 Plus, and connect the DSR terminal input to CTS on the KPC-3 Plus.

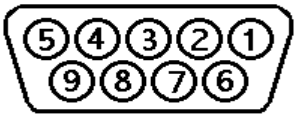
Software settings

When using a standard terminal program to communicate with the KPC-3 Plus, you must set the serial port configuration in your program for 8 data bits, no parity, and one stop bit. The baud supported by the KPC-3 Plus are 300, 600, 1200, 2400, 4800, 9600, and 19200. In general you should use the fastest baud your terminal program and hardware supports. The KPC-3 Plus defaults with ECHO ON, so you should set your terminal program for “full-duplex” operation.

If you are using a baud above 9600 baud and have an older computer, we strongly recommend that you consider installing special high-speed, buffered UARTS in the serial port of your computer. These UARTS are available through many parts suppliers and will improve the reliability of high-speed communications. You should ask for the “high-speed buffered UARTS” (part number 16550 or equivalent) when ordering from your supplier. Newer computers don’t need this enhancement.

Connecting to your Radios

The 1200-baud radio connector on the KPC-3 Plus rear panel is a 9-pin DB style connector, with pin layout as shown below:



The purposes of each pin are as follows:

1 TXA	Transmit audio (AFSK out)
2 XCD	External carrier detect
3 PTT	Push-to-Talk
4 CTLB (AN0)	Control line B (ANØ Analog input)*
5 RXA	Receive audio (AFSK in)
6 GND	Ground
7 EXT-IN	External input for Power/Reset*
8 CTLA (AN1)	Control line A (AN1 Analog input)*
9 GND/RESET	Ground (may be configured as external reset)*

* See the jumper section for details.

Pin 1: Connects to the microphone input of your radio, providing the audio signal to be transmitted.

Pin 2: When a ground is applied to this pin, the KPC-3 Plus will not transmit packets. This pin is normally used as an external means of supplying carrier detect, but can also be used as a “hardware” transmit inhibit.

Pin 3: When the KPC-3 Plus needs to key your transmitter, it will apply a ground to this pin. This is an open-drain circuit and requires a positive voltage from your radio (not to exceed 50 V dc or 200 mA).

Pin 4: This pin is the CONTROL B output from the KPC-3 Plus which may be used as a control line to your radio or other devices. It is an open-drain circuit, and will apply a ground on this pin when controlled by the CTRL command in your KPC-3 Plus (not to exceed 50 V dc or 200 mA). This pin can also be used as AN0 (Analog Input). See the Jumper section for details.

Pin 5: Receive audio from your radio connects to this pin. This normally will connect to the external speaker jack on your radio, but may connect to the receive audio pin on your microphone connector (if available).

Pin 6: This pin connects to the ground of your radio mic connector.

Pin 7: This pin is not normally connected in the KPC-3 Plus, but may be configured as an external reset or to supply power to the KPC-3 Plus using jumpers J6 and J7. See the Jumper section for details.

Pin 8: This pin is the CONTROL A output from the KPC-3 Plus which may be used as a control line to your radio or other devices. It is an open-drain circuit, and will apply a ground on this pin when controlled by the CTRL command in your KPC-3 Plus (not to exceed 50 V dc or 200 mA). This pin can also be used as AN1 (Analog Input). See the Jumper section for details.

Pin 9: This pin is connected to ground as shipped from the factory. You may re-configure this pin to perform an external reset by using jumper J5. See the Jumper section for details.

Interfacing Hand-Held Radios

This section is for those who need more information than that given in the “Installation” chapter.

Many transceivers, especially most hand-held models, obtain Push-To-Talk keying by completing a circuit between the mic input and PTT ground. A direct PTT input to the mic input line of units with this type electret condenser microphone is not usable without some type of isolation.

If you plan to operate with a hand-held transceiver, the KPC-3 Plus has incorporated an isolation circuit which is available by positioning jumper J9 on the center post and the left post (labeled HT) as you face the front of the KPC-3 Plus. Should you later use a different type radio, this change may need to be reconfigured by placing the J9 jumper on the center post and the right post. Most other radios of current manufacture will not require any modification of the KPC-3 Plus.

You may also interface to a hand-held without performing this modification by incorporating the same type of circuitry in the cable from your KPC-3 Plus to your hand-held. Ground return and speaker audio are both supplied through the external speaker jack of your hand-held. While you will need to check the exact requirements of your hand-held radio, the following guidelines may be useful:

- A 0.1 μF capacitor (non-polarized) usually is required in the transmit audio line from the TNC to any handheld transceiver (HT) radio. Even if a capacitor is not required for connection to a particular radio, it can still be installed, and the transmit audio (packets) will pass through to the radio.

- ICOM HT radios key the PTT by providing a low impedance path from the mic input to ground. To accomplish this, simply install a resistor (approximately 3.9 kΩ seems to be a good value) in series with the PTT wire from the KPC-3 Plus, and connect this to the mic input along with the AFSK line.

- The ICOM 2A style wiring is most common, and can be used with some Alinco, Azden, and standard HTs.

- YAESU radios are similar but use a mono plug and a different value resistor.

- The wiring for Yaesu HTs can also be used with Radio Shack HTs.

- Most KENWOOD HT radios key the PTT line by connecting the sleeve of the mic connector to the sleeve of the earpiece connector. This means that you will not need a resistor in the PTT wire from the KPC-3 Plus, simply connect the PTT wire to the sleeve of the mic connector. Another point to watch: most of the KENWOOD HTs (2600 and later) use a three pin mic connector. The AFSK from the KPC-3 Plus should therefore connect to the RING and not the TIP of the mic connector.

- Leave J9 in normal (not HT) when using Kenwood HTs.

Optional Connections to DSUB-9 Radio Port

Pin 7 of the DSUB-9 connector may be configured to allow you to apply power to, or to perform a soft reset on, your KPC-3 Plus.

Applying Power through the DSUB-9 connector.

To power the KPC-3 Plus from Pin 7 of the DSUB-9 connector, you must connect J7 center pin and pin 2, J6 center pin and pin 2.

Resetting the KPC-3 Plus through the DSUB-9 connector.

To perform a soft reset from Pin 7 of the DSUB-9 connector, you must connect J7 center pin and pin 2, J6 center pin and pin 1. Momentarily applying a ground to pin 7 will then perform a soft reset.

Alternatively, pin 9 can be used for either ground or reset with jumper J5: Connect J5 center pin and pin 1 for ground, center pin and pin 2 for external reset.

Appendix B: Advanced Information

Assembly and Disassembly

Should you require access to the KPC-3 Plus to reposition jumpers or for other purposes, disassemble as follows:

1. Turn off power to your KPC-3 Plus and remove all cables from the unit.
2. Using a small phillips screwdriver, remove the two (2) case screws (located on each side of the case) completely.
3. Remove the top cover from the unit. NOTE: In most cases, no further disassembly of the unit is necessary. Only if you need to change a wire jumper or solder connections to the PC board should you continue with disassembly of your KPC-3 Plus.
4. Remove the four (4) hex nuts on the rear panel that secure the DSUB-9 and DSUB-25 connectors to the case.
5. Remove the two (2) screws securing the front panel to the case.
6. Remove the front panel.
7. Observe proper static protection precautions and gently lift the front edge of the PC board to clear the front of the case.
8. Pull the PC board forward to remove the PC board from the case.

To reassemble, reverse the procedure above.

Hard Reset

The hard reset process is provided to re-initialize the KPC-3 Plus to its default values. This process may become necessary should operational problems be encountered or when upgrading your firmware to a new version. The readout specified in step 5 below will be legible only if your terminal baud is 1200. At other terminal baud, a reset will occur. However, no display readout will be observed. This procedure is performed as follows:

1. Open the KPC-3 Plus by removing the two (2) case screws on the sides of the unit and lifting the cover.
2. Locate the Hard Reset jumper (J11). Jumpers are appropriately labeled on the PC board. Refer to the parts location diagram for help in locating them.
3. Place the jumper on both pins.
4. Apply power to the KPC-3 Plus.

5. Observe on the computer display (your terminal program must be set at 1200 baud):

   CHECKSUM OK
   RAM OK
   100,000 BYTES
   NO CLOCK
   REPLACE TEST JUMPER

6. Turn power off.

7. Return jumper J11 to the normal position (one pin only).

8. Reassemble the KPC-3 Plus and return to operation.

Note: After a hard reset, all parameters are at factory default values and any mailbox messages are lost.

Calibration/Equalization

The CALIBRAT command is used to assist in adjusting the data transmitter drive level and also in adjusting the receive equalization of your KPC-3 Plus. The procedures listed below are a guideline for using the calibration command to adjust for optimum performance.

When you issue the CALibrate command, the following options will appear on your screen:

CALIBRATE MODE:
M send mark
R receive
S send space
T send square wave
- or + adjust XMITLVL while transmitting
X exit

Pressing M will transmit a constant MARK (signal) until another key is pressed.

Pressing R will measure a received square wave.

Pressing S will transmit a constant SPACE until another key is pressed.

Pressing T will transmit a square wave (space/mark) until another key is pressed.

Pressing X will exit the calibrate mode and reset the KPC-3 Plus.

Drive level.

The drive level is set by a digital pot, providing a range of 1 mV to 4 V p-p. Default drive level is about 50 mV, corresponding to a XMITLVL command setting of 100. After entering the calibrate mode (CAL) you should use the T option to initiate the transmission of a square wave and then adjust the AFSK drive level by pressing the ‘-’

or '+' key (and holding it down) until you achieve the drive level you need. Adjust deviation to achieve about 3 to 3.5 kHz deviation from your radio.

Equalization.

If you have another Kantronics user nearby, have that station transmit a calibrate square wave and choose the R option in your calibrate mode to measure the status of receive equalization. You will see two numbers displayed each time you press R. The KPC-3 Plus is measuring the space/mark square wave generated by the other station. For best performance, these numbers should be about equal. A 40/60 to 60/40 ratio is acceptable. Change jumper J4 if need be. Default is no equalization, since it's normally not needed.

Note: If you are using a High Impedance Data output from the radio, set J3 on one pin only (OFF = 10 kΩ, ON = 620 Ω).

PTT (Push-to-Talk) Watchdog Timer

The KPC-3 Plus has a built-in PTT Watchdog Timer. The PTT output will be disabled if the TNC has been continuously transmitting for more than approximately 3 min.

To disable the Watchdog timer, jumper the pins of J12.

Microprocessor Watchdog Timer

The TNC includes a watchdog timer within the microprocessor. In the unlikely event that your TNC firmware gets into an infinite loop, the watchdog timer will detect the failure and force a reset of the TNC.

A/D Converter

Two A/D converter inputs are available to the KPC-3 Plus and are used with the Analog command. These inputs can be used to measure dc external voltages from 0 V to 5 V dc with up to 8-bit accuracy.

Note: If you want to measure voltages greater than +5 V dc, the signals need to be attenuated (by installing resistors R13 or R27 on the printed circuit board) to divide them down in voltage. Choose resistance values to scale the desired input voltages to 5 V dc max at test points "J" or "K" on the printed circuit board.

It is possible to bring external voltage into the onboard A/D converters from either the Radio Port or the Serial Port using Jumpers J8 or J10. (See the "jumpers" section.)

So, if you jumper the A/D inputs via this port (instead of the RS232 port) you cannot utilize the A/D inputs and control line outputs at the same time.

- Note: Maximum input measurement can be made. Maximum input voltages to the A/D inputs should not exceed 50 V or damage may occur.

KPC-3 Plus Jumpers

Jumper Locations

- To locate jumpers listed below, use the schematic shipped with the unit and (for locations on the printed circuit board) the parts layout in Appendix E.

Jumper #	Printed Schematic Location	Circuit Location	Name .
J1	A-6	D-4	battery option
J2	A-6	D-4	battery option
J3	D-2	D-1	modem input impedance
J4	D-2	D-1	equalization
J5	C-1	D-2	ground/reset
J6	B-1	D-2	reset/input
J7	B-1	D-2	serial configuration
J8	B-1	D-2	analog input AN0
J9	C-1	D-1	normal/HT
J10	B-2	D-2	analog input AN1
J11	C-4	C-3	reset jumper
J12	C-3	C-1	PTT watchdog
J13	C-6	C-3	ROM size (KPC3-Plus)
J14	C-6	A-3	RAM size
J15	D-4	A-1	NA
J16	A-5	B-4	power supply
J17			full dup (KPC3 Plus MX)
J18			full dup (KPC3 Plus MX)
J19			full dup (KPC3 Plus MX)
J20			full dup (KPC3 Plus MX)

J1 and J2: (battery options) These jumpers can be configured several ways. J1 is normally OFF, and J2 is normally ON. This allows for power from a 9-V battery to be switched into the circuit when the external power plug is removed. If it is desired to have “backup” 9-V battery power in the event of loss of the external power source, J1 should be ON, and J2 should be OFF. In this case the 9-V battery is isolated from the external power by CR2 which prevents a higher external voltage from being applied across the battery.

J3: (modem input impedance) this 2-pin jumper, when installed, sets input impedance of the modem to 600 Ω. When placed on one pin only, input impedance is set to 10 kΩ. Default is 10 kΩ.

J4: (equalization) This two-pin jumper is used to set receive equalization for the modem. When placed on both pins, no equalization is selected, and when placed on one pin, partial equalization is used. See the calibration/equalization section for details. Default is no equalization.

J5: (DSUB-9 Ground/reset) This three-pin jumper connects pin 9 of the radio port to ground or to the reset circuit. When placed on the center pin and pin 1, pin 9 of the DSUB-9 connector is connected to ground. When placed on the center pin and pin 2,

pin 9 is connected to the reset circuit. This allows you to perform a soft reset by applying a momentary ground to pin 9. Default: ground.

J6: (Reset/input) This three-pin jumper, in conjunction with jumper J5 or J7 provides an alternate means to supply dc power to the unit or to perform a reset without powering down the unit. See the section on Connector Configuration for more details. Default: not connected.

J7: (Serial Configuration) This three-pin jumper, in conjunction with J6, provides an alternative means to supply dc power to the unit or to perform a reset without powering down the unit. See the “Optional Connections” section for more details.

Default: center pin and pin 2.

J8: (analog input AN0) When on center pin and pin 1, selects input from DSUB-9 pin 4. When placed on center pin and pin 2, selects input from DSUB-25 pin 18.

J9: (NOR/HT) When placed on the center pin and the pin marked NOR, this configures the AFSK output line for use with base station radios. When placed on the center pin and the pin marked HT, this configures the AFSK output line to also serve as the PTT line for HT radios. Default: NOR.

J10: (Analog input AN1) When on center pin and pin 1, selects input from DSUB-9 pin 8. When placed on center pin and pin 2, selects input from DSUB-25 pin 11.

J11: (Reset Jumper) When on one pin no action occurs. When placed on both pins (after turning unit power off), the processor will perform a “hard” reset (and internal self-test) once power is turned back on. All command parameters will be set to factory default values. Default: not connected.

J12: (PTT watchdog) This jumper enables/disables the PTT watchdog timer. When placed on one pin, the watchdog timer is active; when placed on both pins, the timer is disabled. Default: enabled.

J13: (ROM size) This three-pin jumper permits the installation of different size ROMS. When placed on the center pin and 1, ROMs up to/including 1 megabits may be installed. When placed on the center pin and 2, ROM sizes 1 megabits and above can be installed. Default: 1 megabit and above.

J14: (RAM size) This three-pin jumper allows the installation of various size static RAMs. When placed on the center pin and pin 1, the KPC-3 Plus can accept 32 kbyte or 128 kbyte static RAM. When placed on the center pin and 2, the unit accepts a 512 kbyte static RAM. Default 32K/128K. (does not apply in SMT version, 512K is the default RAM size)

J16: (Power supply) Internal regulator voltage select (5 V/3.3 V) Default 5 V. For details, see the section in this appendix on very low power operation.

J17: MX fulldup option. (does not apply in SMT version)

J18: MX fulldup option. (does not apply in SMT version)

J19: MX fulldup option. (does not apply in SMT version)

J20: MX fulldup option. (does not apply in SMT version)

Appendix C: Options for the KPC-3 Plus

Low Power Operation

Your KPC-3 Plus is capable of very low power operation. In order to achieve the lowest power consumption possible, you must turn the front panel LEDs off with the command LEDS OFF. Also, the modem must be set for INTERNAL carrier detection with the command CD INTERNAL. In this configuration, the KPC-3 Plus will draw less than 15 mA when no signal is being received. Note, however, that the actual current drain may vary slightly depending on the RS232 connections to the unit. Some computers will cause the current drain to be slightly higher than other computers.

If you use software carrier detection (CD SOFTWARE) the current drain will be higher in the “idle” mode since the processor is actively looking for data even when no signal is present.

Very Low Power Operation

KPC-3 Plus circuitry is powered from a linear low dropout regulator at 5.0 V nominal. At this voltage, current requirements vary from approximately 16 mA (unit idle and LEDs OFF) to 23 mA (unit transmitting and LEDs ON). By placing a jumper on J16, the output of the voltage regulator is reduced from 5.0 V to 3.3 V nominal, thus reducing power requirements nearly in half. The current requirements in this case vary from approximately 8.5 mA (unit idle and LEDs OFF) to 12 mA (unit transmitting and LEDs ON).

The 3.3 V mode of operation has not been thoroughly tested in the field and is not guaranteed. The unit may not function properly over extreme temperature ranges when operating at 3.3 V.

Expanding the RAM in the KPC-3 Plus

The KPC-3 Plus is supplied with 128 KiB of RAM. It is possible to expand the RAM in the KPC-3 Plus from 128 KiB to 512 KiB.

Expanding the RAM to 512 KiB requires that you: (1) remove the existing 128 KiB RAM in socket XU14, (2) change jumper J14 to the center pin and pin 2 (to left of the RAM chip), then (3) install a 512 KiB low-power static RAM in XU14, with the pin 1 end of the IC toward J14.

NOTE: in the latest circuit board revision, components are SMT type, and the maximum size 512K static RAM is standard.

Installing the Optional Real-time Clock Module

The Real-time clock option for the KPC-3 Plus consists of a FOX 72421 self-contained clock chip.

Install the clock chip with pin 1 toward the right (as you are looking at the unit) in socket U15.

You will need to initialize the clock with the current date and time after installing the clock chip. Just enter the DAYTIME as described in the commands section of this manual. The Real-time clock is read when the KPC-3 Plus is powered on, or when a soft reset occurs.

Replacing the Lithium Battery

To replace the internal Lithium battery, remove the cover from your KPC-3 Plus and carefully slip the battery out of the battery holder. You may need to use an insulated tool to assist in removal of the battery. When installing a replacement battery (CR2032 or equivalent), be sure you have the positive terminal against the top clip of the battery holder.

NOTE: When you remove the lithium battery from your KPC-3 Plus, all stored parameters will be lost and the unit will revert to the factory default parameters.

Appendix D: In Case of Difficulty

Your Kantronics KPC-3 Plus is manufactured to very stringent quality standards. If you have followed the installation procedures outlined in this manual, it is highly unlikely that you will encounter a failure. If you do have difficulty, use the procedures described in this section to assist in isolating and correcting the problem.

KPC-3 Plus Does Not "Sign-On" to Computer

1. Carefully recheck cabling between your computer serial port and the KPC-3 Plus.

2. Check carefully to insure that the Transmit Data, Receive Data, and Ground leads are connected to the proper pins.

3. If you have made a 5-wire connection to the computer serial port, change to a 3-wire connection. To use HyperTerminal, you need a 5 wire connection, so if you change to a 3 wire connection, you will need to use another terminal communication program.

4. Check your terminal program to be certain it is configured with the correct communications parameters (serial port, baud, parity).

5. Try a "Hard Reset" using the Hard Reset jumper. (Operate your terminal program at 1200 baud when performing a hard reset.)

You Are Unable to Make a "Connect"

1. Issue a connect request and observe the TNC's XMIT LED. If the XMIT LED illuminates, check to insure that the radio is connected to the radio port. If the XMIT LED does not illuminate, check to be sure the XMITOK command is turned ON.

2. Observe the radio to determine if it is being switched to the "Transmit" condition. If not, recheck wiring between the KPC-3 Plus radio port, PTT pin, and ground on the microphone jack.

3. Set the CD command to INTERNAL and turn the radio squelch control to "OFF" and see if the RCV LED illuminates on the packet controller. If it does not light, recheck the audio connection between your transceiver and the KPC-3 Plus.

4. If possible, monitor your transmitted signal with another radio. If the transmitter is keying to "Transmit" but weak or no audio is monitored, increase AFSK output as necessary using R13. (See the AFSK Output Level section.)

Cannot Transmit

1. Check the XMITOK command – it must be ON to allow transmitting.

2. If using a “fixed level” (i.e., the RCV LED stays on all the time) Receive Audio from the radio, which the volume and squelch do not change, set the CD parameter to SOFTWARE.

Cannot Return to Command Mode

The single most common cause of this is that the STOP character (and usually XOFF) have been inadvertently set to the same as the COMMAND character. This is usually caused by the use of the dollar sign ($) as a streamswitch. If you use the $, be aware that you cannot enter hex values without PASSing the dollar sign. Symptoms for this usually are that you can talk to the KPC-3 Plus fine in Command Mode, you can usually talk to others on the air, but you just can’t get back to Command Mode. With most PC terminal programs, pressing <Ctrl+C> will display the heart character, but you still don’t get the cmd: prompt.

Appendix E: Additional Information

Specifications

Size: 0.8" x 5.2" x 5.2" (21 mm x 133 mm x 133 mm)

Weight: 11 oz (0.32 kg)

Power Requirements:

- 6 V dc to 25 V dc, less than 30 mA (LEDS on, unit active)
- 6 V dc to 25 V dc, less than 15 mA (LEDS off, unit inactive)

Power Plug Polarity: Center pin positive

Watch Dog Timer: Approximately 2-1/2 minutes

External Carrier Detect (XCD): Pulldown to ground

External Reset: Pulldown to ground

PTT Output: Open drain, +50 V dc max, 200 mA max

Audio Output:

- Continuously adjustable: 1 mV p-p to 4 V p-p
- Output Impedance (ac coupled): 600 Ω

Modulation:

- 1200 bps FSK, full duplex
- CCITT V.23 (1300 Hz / 2100 Hz)

Audio Input:

- Input Sensitivity: 5 mV p-p
- Dynamic Range: 70 dB
- Input Impedance (unbalanced): 10 kΩ (600 Ω with Jumper J3 installed)
- Max Input Voltage: ±12 V dc; 35 V p-p sinusoidal

Modes of Operation Packet, WeFax, KISS, XKISS, Host, GPS

Other Features: PBBS, KA-NODE, Remote Access, and, as an option, K-Net

Messages from the KPC-3 Plus

The following are brief explanations of the most important messages that may be sent by your TNC.

Note: Since these messages are also sent by other Kantronics TNCs, they refer to "TNC", not to "KPC-3 Plus".

*****(callsign) busy**

The packet station you were attempting to connect to (callsign) is unable to accept connects, and responded with a busy signal.

Already connected on stream n

You are attempting to connect to someone you are already connected to on another stream. The STATUS command will show you who you are connected to and on what stream (i.e., stream "n").

BBS BUSY

You have attempted to connect to your own PBBS, but the PBBS does not have any available streams (PBUSERS) to allow a connect.

CALIBRATE MODE:
M send mark
R receive
S send space
T send square wave
- or + adjust XMITLVL while transmitting
X exit

This message appears on your screen when you enter the Calibrate Mode, and it prompts you to press M to generate a MARK tone, R to receive a Kantronics Calibrate signal, S to generate a SPACE tone, T to transmit a Kantronics Calibrate signal, - or + to adjust XMITLVL while transmitting, or X to return to the Command Mode.

Can't DISCONNECT

You are not connected on this stream so therefore cannot disconnect. This message will be followed by the stream and a "Link state is:" message, described later in this section.

Can't RECONNECT

You have attempted to reconnect to a station (by issuing a CONNECT command) but the callsign you entered is not the same as the station you are already connected to. You may only reconnect to the station you are connected to on this stream.

CHECKSUM ERROR

This message indicates that the Kantronics firmware in your TNC may be damaged. You may see this message when performing a hard reset, or any time a soft reset is performed (including initial power up), and an EPROM error is detected.

CHECKSUM OK

When performing a hard reset, this message indicates that the Kantronics firmware EPROM has passed the internal checksum test.

cmd:

This is the Command Mode's prompt for input. Any characters entered after the TNC prints "cmd:" will be used as command input and not packet data.

Command not available in NEWUSER mode

This message indicates that you have attempted to change one of the commands in the TNC, but that command is not currently available to you. This occurs when the INTFACE command is set to NEWUSER. To enable all commands, set INTFACE TERMINAL.

***** connect request:**

A remote packet station has attempted to connect to you, but there is not a valid stream available for the connection. The remote station will be sent a busy message packet. See the USERS and MAXUSERS commands for setting more streams and allowing more connects at one time if desired. Also be sure CONOK is ON.

***** CONNECTED to call [VIA digi1....digi8]**

A packet connection has taken place. This can happen by you issuing a connect request or a connect request coming in from a remote station. "call" will be the callsign entered in the remote stations MYCALL and if a path was used it will be shown.

***** DISCONNECTED**

The packet connection no longer exists.

EH?

This is the TNC's generalized "I don't understand" message. A dollar sign ($) is used to point to the offending character. It will also appear if a required input item is missing, e.g.:

```
C KV7B V
         $
EH?
```

In this example, the required callsign after the VIA option is missing. Most commands that receive an EH? error are ignored. In a few cases, part of the command may be accepted and acted upon, as described under the message "Input ignored".

ENTER YOUR CALLSIGN=>

Type in your callsign. The TNC needs to know who you are to properly implement its protocol. Your callsign will be placed in the MYCALL parameter and used to generate the MYNODE and MYPBBS callsigns, and the CWIDTEXT.

xxxx FREE BYTES

This message indicates how many additional characters may be entered into the TNC packet buffer before the buffer is full.

*****FRMR received:**

A frame reject packet has been received for an unknown reason. The information field of this packet will display in hexadecimal value. This display may be useful in determining why the receiving station rejected your packet.

*****FRMR sent:**

Frame reject packet has been sent due to a detected error in protocol. Three bytes (6 hexadecimal characters) are displayed to assist in determining the reason for the reject.

Input ignored

Since the TNC command interpreter was kept small and simple, sometimes it will change parameters before it completes parsing some of the more involved commands. In some cases, options at the beginning of the command will have been acted on before a syntax error near the end of the line is reached. When this occurs, “Input ignored” is used to show what part of the line was ignored. The dollar sign points to the boundary: characters to the left were used; the character pointed to and those to the right were not, i.e., the line was parsed as if a <CR> was entered at the $. For example:

```
BUDLIST QST,KBØNYK NØKN
$
Input ignored
```

Because the comma is missing, the command is interpreted as if it were BUDLIST QST,KBØNYK; the NØKN is ignored.

INVALID STREAM

The stream you have tried to change to is not valid. Stream designators must be letters A - Z. MAXUSERS sets the upper limit on valid streams. The setting of LCSTREAM determines if lower-case characters are permitted when switching streams.

KANTRONICS KPC3PMX VERSION 9.1
(C) COPYRIGHT 1988-2007 BY KANTRONICS INC. ALL RIGHTS RESERVED.
DUPLICATION PROHIBITED WITHOUT PERMISSION OF KANTRONICS.

A message such as this (different for each product, but with this format) appears when the TNC is first turned on and after any soft reset, including changing the NUMNODES, PBBS, MAXUSERS or MYREMOTE commands, or issuing the RESET command.

Link state is:

This message is output in response to the CONNECT and DISCONNECT commands if the state of the link does not permit the requested action. It is prefaced by “Can’t CONNECT” or “Can’t DISCONNECT” as appropriate and will be followed by the current link state. A CONNECT command with no options will display the link state of the current stream.

Current link states are:

Both devices busy

Both TNCs involved in the connection are unable to accept any more data.

CONNECTED to (callsign v path)
Your TNC is currently connected to the indicated station, using the path given.

CONNECT in progress
Your TNC is attempting to establish a connection.

Device busy
Your TNC is unable to accept any more data from the remote station at this time.

DISC in progress
Your TNC is attempting to disconnect from another station.

DISCONNECTED
No connection exists on the current stream.

FRMR in progress
Your TNC has detected an error in the protocol. This is normally caused by two TNCs using the same callsign, resulting in both of them trying to respond to the same message.

Remote device busy
The remote TNC is unable to receive any more data from the radio at this time.

Waiting ACK and device busy
Your TNC has sent a packet to another station and is waiting for the acknowledgment, but your TNC is not able to accept any data from the radio at this time.

Waiting ACK and remote busy
Your TNC has sent a packet to another station and is waiting for the acknowledgment, and the remote TNC is not able to accept any data from the radio at this time.

Waiting ACK and both devices busy
Your TNC has sent a packet to another station and is waiting for the acknowledgment, but neither your TNC nor the other station's TNC is able to accept any data from the radio at this time.

Waiting acknowledgment
You have sent a packet of data to another station, and your TNC is waiting for the acknowledgment.

MESSAGES WOULD BE LOST
This message indicates that you have attempted to set the PBBS size too small to hold all of the existing messages in the mailbox. The size is not changed, and therefore no messages are lost. If you want to make the mailbox smaller, you must first delete some of the messages or set the PBBS to 0, erasing all of the messages, then set the new PBBS size.

NO KNOWN NODES

An NDHEARD list has been requested and the TNC has not heard any other nodes (e.g., KA-Nodes).

NOT ENOUGH RAM

This message indicates that you have attempted to set the NUMNODES, MAXUSERS, or PBBS to a value, which requires more RAM memory than is currently available. It may also occur if you try to enable the MYREMOTE and there isn't enough memory left.

Not while connected

Some parameters cannot be changed if the TNC is connected to another station. This message is printed if an attempt is made.

PBBS MESSAGE BUFFER NOT VALID!
TRYING TO RECOVER

When you turn on the TNC a check is made of the pointers relating to the PBBS. It appears that one or more of these pointers may not be correct. An attempt is made to correct them, but it is possible that some or all of the messages may be lost.

PRESS (*) TO SET BAUD

You need to press the asterisk (*) key on your keyboard within 2 s of seeing this message. The KPC-3 Plus' autobaud routine will then detect what baud your computer is using, so the two devices can communicate.

RAM OK xxxK BYTES

This message appears when a hard reset is performed and indicates that the TNC has tested the installed RAM and has found no problems. It also reports the amount of RAM installed in your TNC.

RAM ERROR xxxK BYTES

If this message appears when you perform a hard reset, the TNC has detected a problem with the installed RAM, and reports how much of the RAM it found.

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

The number of tries set by the RETRY command has been exceeded. Therefore the connection has been broken.

S00

This message from the TNC indicates that it is in the HOST mode of operation, and has just performed a reset. To exit the Host mode, you must send three characters to the TNC from your keyboard. See the Host mode section of the manual for details. Two FEND characters precede this message and one FEND character follows this message, but they may not be visible (depending on your terminal program).

Value out of range

If the syntax of the command is legal, but the value specified is too large or too small for this command, the value out of range message is returned. A $ is used to point to the bad value.

was

Whenever one of the parameters is changed, the previous value is usually displayed.

Example:

```
cmd:AX25 OFF
AX25L2V2 was ON
```

YOU HAVE MAIL

A new message for you has just been put in the PBBS.

ASCII Chart

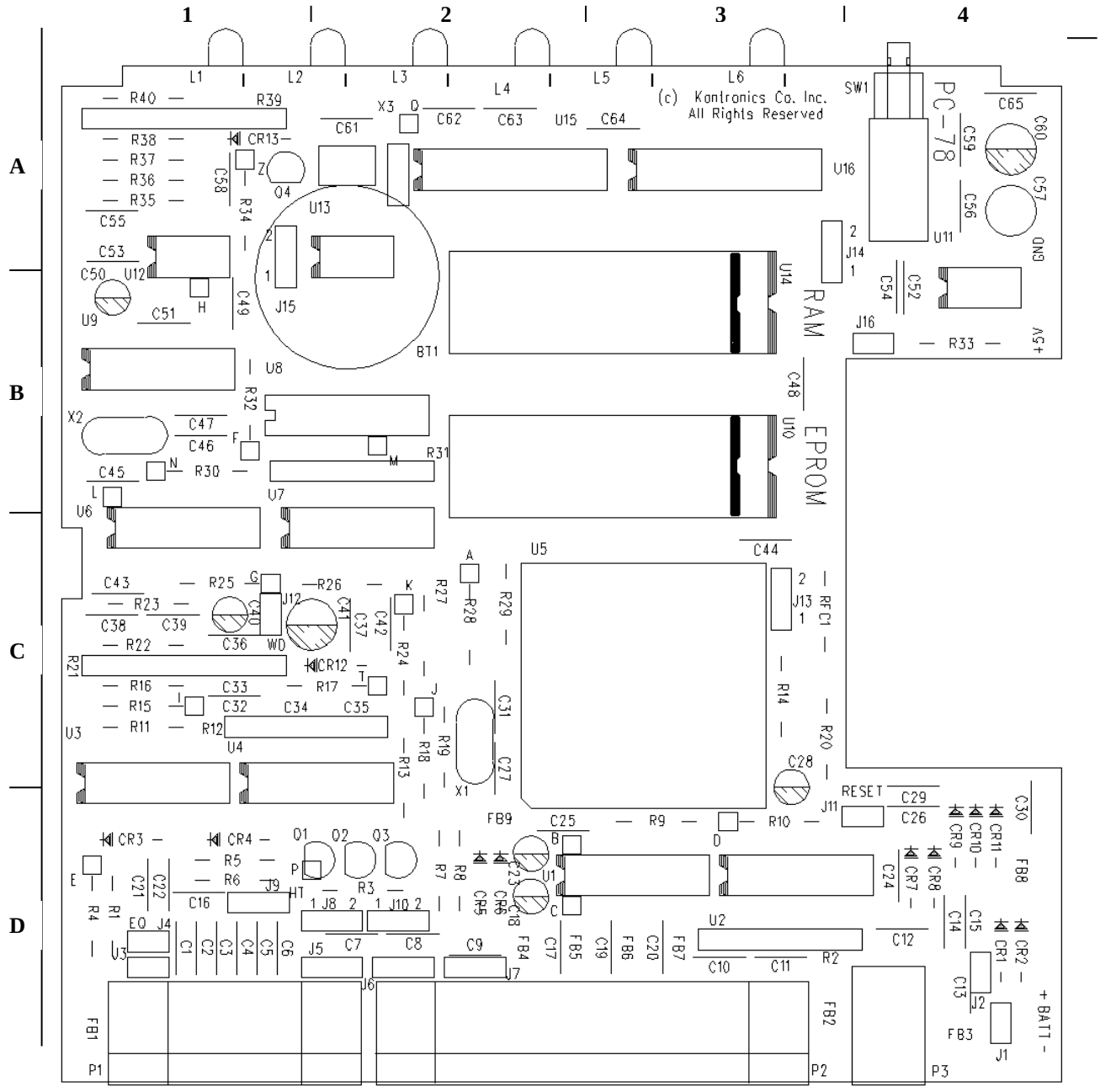
Ctrl	Dec	Hex	Code	Dec	Hex	Code	Dec	Hex	Code	Dec	Hex	Code
@	0	00	NUL	32	20	SP	64	40	@	96	60	`
A	1	01	SOH	33	21	!	65	41	A	97	61	a
B	2	02	STX	34	22	“	66	42	B	98	62	b
C	3	03	ETX	35	23	#	67	43	C	99	63	c
D	4	04	EOT	36	24	$	68	44	D	100	64	d
E	5	05	ENQ	37	25	%	69	45	E	101	65	e
F	6	06	ACK	38	26	&	70	46	F	102	66	f
G	7	07	BEL	39	27	‘	71	47	G	103	67	g
H	8	08	BS	40	28	(	72	48	H	104	68	h
I	9	09	HT	41	29	)	73	49	I	105	69	i
J	10	0A	LF	42	2A	*	74	4A	J	106	6A	j
K	11	0B	VT	43	2B	+	75	4B	K	107	6B	k
L	12	0C	FF	44	2C	,	76	4C	L	108	6C	l
M	13	0D	CR	45	2D	-	77	4D	M	109	6D	m
N	14	0E	SO	46	2E	.	78	4E	N	110	6E	n
O	15	0F	SI	47	2F	/	79	4F	O	111	6F	o
P	16	10	DLE	48	30	0	80	50	P	112	70	p
Q	17	11	DC1	49	31	1	81	51	Q	113	71	q
R	18	12	DC2	50	32	2	82	52	R	114	72	r
S	19	13	DC3	51	33	3	83	53	S	115	73	s
T	20	14	DC4	52	34	4	84	54	T	116	74	t
U	21	15	NAK	53	35	5	85	55	U	117	75	u
V	22	16	SYN	54	36	6	86	56	V	118	76	v
W	23	17	ETB	55	37	7	87	57	W	119	77	w
X	24	18	CAN	56	38	8	88	58	X	120	78	x
Y	25	19	EM	57	39	9	89	59	Y	121	79	y
Z	26	1A	SUB	58	3A	:	90	5A	Z	122	7A	z
[	27	1B	ESC	59	3B	;	91	5B	[	123	7B	{
\	28	1C	FS	60	3C	<	92	5C	\	124	7C	|
]	29	1D	GS	61	3D	=	93	5D	]	125	7D	}
^	30	1E	RS	62	3E	>	94	5E	^	126	7E	~
_	31	1F	US	63	3F	?	95	5F	_	127	7F	DEL

KPC-3 Plus Parts List

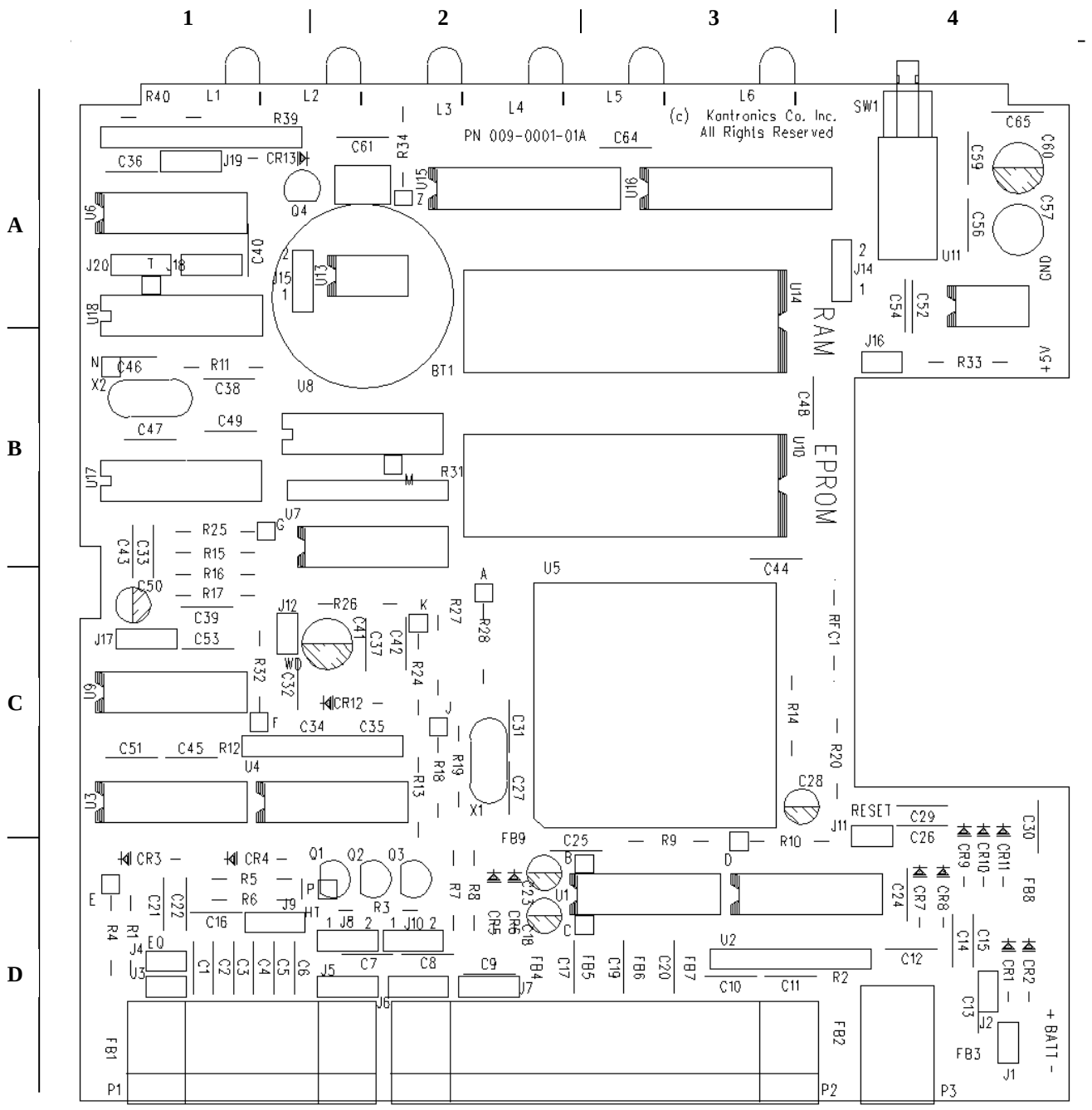
BT1	2032	C43	.1uF	J6	3P_SIH	R20	4.7 kΩ
C1	.001uF	C44	.1uF	J7	3P_SIH	R21	100K_X5_SIP
C2	.001uF	C45	.1uF	J8	3P_SIH	R22	150 kΩ
C3	.001uF	C46	33pF	J9	3P_SIH	R23	3.9 kΩ
C4	.001uF	C47	33pF	J10	3P_SIH	R24	10 kΩ
C5	.001uF	C48	.1uF	J11	2P_SIH	R25	1 MΩ
C6	.001uF	C49	.1uF	J12	2P_SIH	R26	3.3 MΩ
C7	.001uF	C50	1uF	J13	3P_SIH	R27	
C8	.001uF	C51	180pF	J14	3P_SIH	R28	10 kΩ
C9	.001uF	C52	.001uF	J15	3P_SIH	R29	100 kΩ
C10	.001uF	C53	.1uF	J16	2P_SIH	R30	1 MΩ
C11	.001uF	C54	.1uF	L1-LED_YELLOW_T1		R31	100K_X4_SIP
C12	.001uF	C55	180pF	L2-LED_GREEN_T1		R32	620 Ω
C13	.001uF	C56	.001uF	L3-LED_GREEN_T1		R33	180 kΩ
C14	.1uF	C57	10uF	L4-LED_GREEN_T1		R34	620 Ω
C15	.1uF	C58	.01uF	L5-LED_RED_T1		R35	82 kΩ
C16	.0022uF	C59	.1uF	L6-LED_GREEN_T1		R36	270 kΩ
C17	.001uF	C60	10uF	P1	DB9 Female	R37	3.3 kΩ
C18	1uF	C61	.1uF	P2	DB25 Female	R38	9.1 kΩ
C19	.001uF	C62	22pF	P3	2.1 mm jack	R39	1.5K_X5_SIP
C20	.001uF	C63	22pF	Q1	2N7000	R40	1.5 kΩ
C21	.01uF	C64	.1uF	Q2	2N7000	RFC1	10uH
C22	10pF	C65	.1uF	Q3	2N7000	SW1-PHA012U10EEM	
C23	1uF	CR1	1N914	Q4	PN2907A	U1	14C88
C24	.001uF	CR2	1N914	R1	620 Ω	U2	74HC14
C25	.001uF	CR3	1N914	R2	100K_X4_SIP	U3	LMC6034IN
C26	.1uF	CR4	1N914	R3	4.7 kΩ	U4	74HC14
C27	22pF	CR5	1N914	R4	10 kΩ	U5	MC68HC11F1
C28	1uF	CR6	1N914	R5	10 MΩ	U6	73M223
C29	.1uF	CR7	1N914	R6	47 kΩ	U7	74HC00/
C30	.1uF	CR8	1N914	R7	10 kΩ	U8	74HC138
C31	22pF	CR9	1N914	R8	10 kΩ	U9	AD8402
C32	.1uF	CR10	1N914	R9	10 kΩ	U10	ROM_32
C33	.1uF	CR11	1N914	R10	47 kΩ	U11	MIC2951
C34	.001uF	CR12	1N914	R11	330 kΩ	U12	LMC6032IN
C35	.001uF	CR13	1N6263	R12	100K_X4_SIP	U13	X25128
C36	470pF	FB1-9	SM FERRITE	R13		U14	RAM_32
C37	.1uF	BEAD		R14	1 MΩ	U15	72421
C38	.1uF	J1	2P_SIH	R15	6.8 kΩ	U16	74HC374
C39	.01uF	J2	2P_SIH	R16	150 kΩ		
C40	1uF	J3	2P_SIH	R17	10 kΩ	X1	XTAL
C41	47uF	J4	2P_SIH	R18	10 kΩ	X2	XTAL
C42	.001uF	J5	3P_SIH	R19	10 MΩ	X3	32.768KHZ

KPC-3 Plus MX Parts List

BT1	2032
C1	.001uF
C2	.001uF
C3	.001uF
C4	.001uF
C5	.001uF
C6	.001uF
C7	.001uF
C8	.001uF
C9	.001uF
C10	.001uF
C11	.001uF
C12	.001uF
C13	.001uF
C14	.1uF
C15	.1uF
C16	.0022uF
C17	.001uF
C18	1uF
C19	.001uF
C20	.001uF
C21	.01uF
C22	10PF
C23	1uF
C24	.001uF
C25	.001uF
C26	.1uF
C27	22pF
C28	1uF
C29	.1uF
C30	.1uF
C31	22PF
C32	.1uF
C33	.1uF
C34	.001uF
C35	.001uF
C36	.1uF
C37	.1uF
C38	100pF
C39	180pF
C41	47uF
C42	.001uF
C43	.1uF
C44	.1uF
C45	.1uF
C46	18pF
C47	18pF
C48	.1uF
C49	.1uF
C50	1uF
C51	180pF
C52	.001uF
C53	.1uF
C54	.1uF
C56	.001uF
C57	10uF
C59	.1uF
C60	10uF
C61	.1uF
C62	22pF
C63	22pF
C64	.1uF
C65	.1uF
CR1	1N914
CR2	1N914
CR3	1N914
CR4	1N914
CR5	1N914
CR6	1N914
CR7	1N914
CR8	1N914
CR9	1N914
CR10	1N914
CR11	1N914
CR12	1N914
CR13	1N6263
FB1-9	SM FERRITE BEAD
J1	2P_SIH
J2	2P_SIH
J3	2P_SIH
J4	2P_SIH
J5	3P_SIH
J6	3P_SIH
J7	3P_SIH
J8	3P_SIH
J9	3P_SIH
J10	3P_SIH
J11	2P_SIH
J12	2P_SIH
J13	3P_SIH
J14	3P_SIH
J15	3P_SIH
J16	2P_SIH
J17	3P_SIH
J18	3P_SIH
J19	3P_SIH
J20	3P_SIH
L1-LED_YELLOW_T1	
L2-LED_GREEN_T1	
L3-LED_GREEN_T1	
L4-LED_GREEN_T1	
L5-LED_RED_T1	
L6-LED_GREEN_T1	
P1	DB9 Female
P2	DB25 Female
P3	2.1 mm jack
Q1	2N7000
Q2	2N7000
Q3	2N7000
Q4	PN2907A
R1	620 Ω
R2	100K_X4_SIP
R3	4.7 kΩ
R4	10 kΩ
R5	220 kΩ
R6	47 kΩ
R7	10 kΩ
R8	10 kΩ
R9	10 kΩ
R10	47 kΩ
R11	100 kΩ
R12	100K_X4_SIP
R13	
R14	1 MΩ
R15	9.1 kΩ
R16	82 kΩ
R17	150 kΩ
R18	10 kΩ
R19	10 MΩ
R20	4.7 kΩ
R24	10 kΩ
R25	100 kΩ
R26	3.3 MΩ
R27	
R28	10K Ω
R31	100K_X4_SIP
R32	620 Ω
R33	180 kΩ
R34	620 Ω
R39	1.5K_X5_SIP
R40	1.5 kΩ
RFC1	10uH
SW1-PHA012U10EEM	
U1	14C88
U2	74HC14
U3	LMC660CN
U4	74HC14
U5	MC68HC11F1
U6	74HC00
U7	74HC00/
U8	74HC138
U9	AD8402
U10	ROM_32
U11	MIC2951
U13	X25128
U14	RAM_32
U15	72421
U16	74HC374
U17	MX614
U18	MX614
X1	XTAL
X2	XTAL
X3	32.768KHZ



KPC-3 Plus Parts Layout



KPC-3 Plus MX Parts Layout