The Amateur Packet Radio Handbook — Chapter 4
Copyright James Wagner, 1993 — INTERMEDIATE OPERATING

Once you have had a few conversations on packet and have started to find your way around your neighborhood network, many questions start to come up.

Chapter 5 attempts to provide you with information about the basic idea of servers; though this name is not frequently used, what it represents pervades our system. Chapters 6, 7, 8 & 9 try to provide working information about bulletin boards, DX spotting, and network operation. Chapter 10 discusses some of the basics of AX.25. Detailed information about bulletin boards and nodes (networking) is found in Volume 2.

In this chapter, we try to explore some of the more general operating questions which frequently come up.

4.A OPERATING HINTS

4.A.1 OPERATING POWER & RETRIES: FCC regulations require us to use the minimum power consistent with reliable com-munications. In the case of packet, reliable communication translates into low retry rate. Increasing power may reduce the number of retries your station takes to deliver a packet. But if the station at the other end requires many retries, you still have a problem. If the other station is a personal station, it may increase power, use a directional antenna, or use a digipeater. But if the station is a hilltop node, there isn't much that can be done at the that end. In this case, you may reduce retries on both ends with a directional antenna (and low power, all at the same time!)

For poor link quality, some simple rules might help. Increasing your power (usually) decreases your outgoing retries but but have no effect on incoming retries. Higher gain antenna should decrease both incoming and outgoing retries. A properly aimed directional antenna should reduce incoming retries; it may reduce outgoing retries unless it makes you a hidden transmitter to other users.

4.A.2 CHANGABLE PROPAGATION: Many areas experience seasonal and weather changes in VHF propagation. At the author's location, several nodes are available. The closest one (about 15 miles, line of sight) is quite variable, changing with weather and season; this may be partially attributable to relatively low power (a few watts). Another node, (about 30 miles, just out of line of sight) is quite reliable but, during heavy use times, is fraught with hidden transmitter problems. When you are unable to connect with your favorite node, don't immediately assume that it is a node problem. If your area is prone to fall temperature inversions (morning fog is a good indicator), this time is often especially difficult for propagation.

Packet tends to be more sensitive to propagation than voice. While you may be able to understand a poor quality voice signal, packet is less tolerant.

4.B KEYBOARD USE

When you type, there are two very significant things to remember. The first has to do with packets sent to your TNC; the second has to do with compatibility with various computers.

4.B.1 UNDOING WHAT HAS BEEN TYPED: Unless you change your TNC parameters, the send packet character is a carriage return (*enter* or *return*). Once this key has been pressed, the characters you have typed are formed into a packet and stored for sending. Unless you use a special cancel packet command (for TNC2s, see *CANPAC*), what you have sent is forever beyond your control. You cannot fix errors, etc, after that point. What is done is done.

Prior to the carriage return, you can backspace to your heart's delight, up to the first character following the preceding carriage return. Author's advice: hurrying or disliking the backspace key is no excuse for poor spelling!

4.B.2 COMPATIBILTY WITH OTHER COMPUTERS: One of the dilemmas is how to keep what is typed compatible with other computers. There are few computers left with 40 character lines (the Apple II was one). There are many with 64 character lines (Commodore C64s among them). Displays with 80 character widths are common but not universal. So what do we do?

If the other person uses a carriage return at the end of each line, you will see exactly what their screen width is. If you do the same, they will see yours. The strategy is for the one with the wider screen to use the carriage return at the same place on their screen as with the incoming packets.

Not all computers recognize all of the characters which can be sent by other computers. A common character in this category is the TAB character. Unless two computers have the same TAB settings, text with TABs will come out strange on the other screen. The simple solution is to just avoid TAB!

Also avoid relying on automatic line-wrap.   Many terminal programs automatically insert carriage returns at the end of the screen, whether you press the key or not. Many others do not. Many terminal programs have problems with text containing very long lines (longer than 256 characters, in particular). This is what happens when you rely on line wrap with a terminal program which does not insert carriage returns. For this reason and the others outlined above, please just use that enter/return key at the right-hand end of each line!

See also Chapter 13, section A, for comments pertinent pre-typed text files.

4.C THAT BLANKETY-BLANK "*** DISCONNECTED"

When using very busy nodes or nodes which link on the same frequency as that the user frequency (see section 8.D), surprise disconnects can be quite frequent. There is one common cause and another less common cause. A disconnect happens when your station or the station at the other end has exceeded its retry limit for your connection.   If the station at the other end (or somewhere else in a circuit of several nodes) retries out, you get:

```
*** DISCONNECTED
```

from your TNC. But if YOUR station retries out, you get:

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

The most common cause is the hidden transmitter situation (see section 3.E.2). One of the problems is that it is quite difficult to tell when it is happening. See section 4.D for information which might let you know when it is happening.

The less common cause is a change in propagation which causes the link to degrade. Possible, but very unlikely, causes are equipment failure at either end or accidental disconnect commands.

4.D WHAT THE TNC MONITOR FUNCTION TELLS YOU

The TNC's monitor capability can let you observe what is happening on your operating frequency. The monitor display can also be pretty confusing. And its format is not the same for all TNCs. To understand what is shown, it is necessary to look first at the several basic packet types.

4.D.1 PACKET TYPES: There are about six basic packet types. Each serves a specific purpose. Connect (C) packets are used exclusively for connect requests; these are also known as SABM packets. Information (I) packets contain characters in the body of the packet. Unnumbered Information (UI) packets are sent out to nobody specific (as in CQs or beacons); these packets can also be thought of as Unconnected Information. UI packets are also the basic communication medium of TCP/IP (See Chapter 10). Disconnect" (DISC) packets are used exclusively for disconnect requests. Disconnected Mode (DM) packets are used to signal that a station is unable to accept a connect request. Unnumbered Acknowledge (UA) packets are used to acknowledge the receipt of either connect or disconnect requests.

The packet types which tell the most about frequency congestion and retries are the RR, REJ and RNR packets. An RR packet (Receiver Ready) is used to signal the error-free receipt of a packet. An RR packet is in response to an I packet. I packets are numbered in the pattern 0,1,2...,6,7,0,1,... ; the RR packet indicates the number of the I packet is is acknowledging.

RNR (Receiver Not Ready) packets are seen most often from nodes or bulletin boards. It means that the station cannot receive any more packets. The usual cause is that packets which must be sent on somewhere else have not been delivered. Thus, they are building up in the station and occupying memory (buffer) space. When the station has run out of space for that conversation, it sends an RNR; when it is able to accept more, it sends an RR.

REJ (Reject) is sent when a packet has been received with an error. It is a request to resend the packet. The REJ packet contains the number of the I packet which had an error.

4.D.2 CONGESTION SYMPTOMS: Congestion is easily seen because packets have both the TO and FROM callsigns in them. Hidden transmitters are particularly obvious when you see only one-half of a conversation (see 4.E because there is more than one reason for seeing only half of a conversation). When there are more than 6-8 stations on a single frequency, even if not hidden, collisions become significant.

4.D.3 POOR LINK SYMPTOMS: Link quality is best observed during a light traffic time. If there are few other users on the frequency and both your station and the station you are in contact with require several retries per packet, the link is poor. Sometimes the excessive retry rate may be one-sided such as when one station has much more transmit power than the other. Retries can be seen by watching the PTT indicator on your TNC; every time it flashes after you press the enter/return key is a TRY. Expect one flash; more are retries.

4.D.4 MONITOR DISPLAY: For starters, lets look at an example of a TNC2-style monitor display of a complete connect & disconnect process.

```
cmd: c lyons
cmd:*** CONNECTED to LYONS
c 1 ka7upd
KA7EHK-15>KA7UPD <C C P>
KA7UPD>KA7EHK-15 <UA R F>
KA7UPD>KA7EHK-15 <I C S0 R0>:Mailbox ready
LYONS:KA7UPD-1} Connected to KA7UPD
KA7EHK-15>KA7UPD <RR R F R2>
Mailbox ready
Mailbox (B,H(elp),K,L,R,S) >
b
KA7EHK-15>KA7UPD <I C P S0 R2>:b
KA7UPD>KA7EHK-15 <D C P>
KA7EHK-15>KA7UPD <UA R F>
*** DISCONNECTED
```

The first 2 lines show a connect to a node. The 3rd line is a connect request for G8BPQ style nodes. The 4th line is the connect request BY THE NODE on behalf of KA7EHK (see section 4.E for an explanation of the SSID change). The 5th line is an acceptance by KA7UPD of the connect request back to the node. This is followed immediately by the mailbox logon message by KA7UPD. Note that this is an I packet which is numbered R0. The next line is (finally) the notification by the node that the connect request was successful. The 8th line with the RR packet is an acknowledgement by the node of successful receipt of the preceding packet(s). The next 2 lines are the response from the KA7UPD mailbox as it actually comes out of KA7EHK's TNC. Note that there is one more line here than monitored between KA7UPD and the node. This packet was probably received with an error at the monitoring site and, thus, not shown. The next line is the bye command to KA7UPD's mailbox from KA7EHK. The I packet numbered R2 from the node (KA7EHK-15) to KA7UPD is the bye command being sent by the node to KA7UPD. That is followed by the disconnect (D) packet as sent by KA7UPD mailbox. That is, in turn, followed by the receipt acknowledgement for the disconnect by the node. The very last line shows the TNC's response to the disconnect from the node.

Next, lets look at a situation where the other station is hidden:

```
cmd:c valley
cmd:*** CONNECTED to VALLEY
c n7fgf
KA7EHK-15>N7FGF <C C P>
VALLEY:N7FGF-3}  Connected to N7FGF
KA7EHK-15>N7FGF <RR R F R1>
[AEA PK-88]  19096 free  (A,B,H,J,K,L,R,S,V,?) >
b
KA7EHK-15>N7FGF <I C S0 R1>:b
KA7EHK-15>N7FGF <UA R F>
*** DISCONNECTED
```

In the preceding case it all goes like the first example except that there are no monitored packets from N7FGF. There are two minor differences: (1) this is a TheNet node and the connect command is a little different than G8BPQ nodes, and (2) the mailbox response has only one line instead of two.

What was just shown in the first two examples are nearly ideal packet transactions. Lets look at a little more normal situation. We will take this in sections to make it a bit simpler. Remember, however, that the next several boxes are part of a single monitoring log of only a few minutes duration.

```
SALEM>N7TRJ <UA R F>
SALEM>N7TRJ <UA R F>
SALEM>N7TRJ <RR C P R0>
SALEM>N7TRJ <I C P S0 R0>:Welcome to the Salem Packet
Switch
Type ? for list of available commands.
SALEM>N7TRJ <RR R F R1>
SALEM>N7TRJ <RR R F R2>
```

From the previous examples, we can interpret these lines pretty easily. The first 2 lines are the node's response to N7TRJ's connect request. Finally, in line 4, the node logon message is sent. This is followed by RR packets acknowledging two packets from N7TRJ. We cannot tell what these packets are because N7TRJ is hidden from the monitoring site.

```
SALEM>KB7NZ <UA R F>
SALEM>KB7NZ <I C P S0 R0>:Welcome to the Salem Packet
Switch
Type ? for list of available commands.
SALEM>N7TRJ <RR R F R3>
SALEM>KB7NZ <UA R F>
SALEM>KB7NZ <RR C P R0>
SALEM>N7TRJ <RR R F R4>
SALEM>KB7NZ <RR R F R0>
SALEM>KB7NZ <RR R F R1>
SALEM>KB7NZ <RR C P R1>
SALEM>KB7NZ <I C P S0 R1>:Welcome to the Salem Packet
Switch
Type ? for list of available commands.
```

In the midst of N7TRJ's node activities, now KB7NZ is making a connect request. This can be determined in line 1 because of the UA packet which is an acknowledgement of a connect request. This is immediately followed by the node logon message to KB7NZ. But not all is well with KB7NZ because a second connect acknowledge appears in line 5. This means that KB7NZ made a second connect request because the first acknowledge was not heard. Then, at the end, the logon message is sent again. Note that this time, it is numbered R1 while the first was R0. Note, also, that when several RR packets are sent to the same station with the same R number, it indicates that the station did not hear the earlier acknowledgement and resent the same packet, thus generating a new RR acknowledgement.

```
SALEM>KB7NZ <I C P S1 R1>:SALEM:AF7S-1}  Connected to
SRABBS:N7IFJ

SALEM>KB7NZ <I C P S2 R1>:[MSYS-1.12-H$]

SALEM>KB7NZ <I C S3 R1>:Hello Bob, Welcome to N7IFJ's MSYS
BBS in Salem, OR

SALEM>KB7NZ <I C P S5 R1>:Enter command:
A,B,C,D,G,H,I,J,K,L,M,N,P,R,S,T,U,V,W,X
,?,* >
```

KB7NZ sent the node a command to connect to SRABBS. We did not see that request, only notification of the connect. The connect has happened on another port (frequency) of the node and is not visible here. This is followed by the BBS logon message to KB7NZ. These packets are sent but we do not see KB7NZ's acknowledge-ment of each because that transmitter is hidden from the monitoring site. Also, there is a missing packet (S4) which was not shown because of errors in the received packet at the monitoring site.

```
SALEM>N7TRJ <I C P S1 R4>:SALEM:AF7S-1}  Connected to
BEND:K7YLO-1
SALEM>N7TRJ <RR R F R5>
SALEM>KB7NZ <RR C P R1>
```

Now, in the first line, we see the node notify N7TRJ that a connect request to the distant node BEND is successful. It appears to be followed in the 3rd line with the acknowledgement of another packet from N7TRJ.

```
SALEM>KD7YB <UA R F>
SALEM>KD7YB <I C P S0 R0>:Welcome to the Salem Packet
Switch
Type ? for list of available commands.

SALEM>N7TRJ <RR C P R5>
KD7YB>SALEM <RR R F R1>
SALEM>KD7YB <RR R F R1>
SALEM>KD7YB <I C P S1 R1>:SALEM:AF7S-1}  Connected to
SRABBS:N7IFJ
```

Here is yet a third connect request to the node, this time by KD7YB. And, like KB7NZ, a connection to SRABBS was requested. This time, however, KD7YB is not hidden from the monitoring site.

```
KD7YB>SALEM <RR R F R2>
SALEM>KD7YB <I C S3 R1>:Hello Joan, Welcome to N7IFJ's MSYS
```

```
BBS in Salem, OR
SALEM>KD7YB <I C S4 R1>:Use  D HELP.DOC to download a help
file (Tnx to KA7ZIN), enjoy.

SALEM>N7TRJ <RR C P R5>
KD7YB>SALEM <REJ R R2>
SALEM>KB7NZ <RR R F R1>
SALEM>KB7NZ <RR R F R2>
SALEM>KD7YB <RR C P R1>
SALEM>KD7YB <I C S2 R1>:[MSYS-1.12-H$]

SALEM>KD7YB <I C S3 R1>:Hello Joan, Welcome to N7IFJ's MSYS
BBS in Salem, OR
SALEM>KD7YB <I C S4 R1>:Use  D HELP.DOC to download a help
file (Tnx to KA7ZIN), enjoy.

SALEM>KD7YB <I C P S5 R1>:You have unread mail, please kill
when read:

SALEM>KB7NZ <RR C P R2>
SALEM>KD7YB <I C P S0 R1>:Enter command:
A,B,C,D,G,H,I,J,K,L,M,N,P,R,S,T,U,V,W,X
,?,* >

KD7YB>SALEM <RR R F R1>
```

The preceding group of monitored packets includes a REJ. This means that the node received a packet from KD7YB and that packet contained an error. The REJ packet is a request to resend the packet with the error. It is difficult from this point to associate which packet had the error and that is not so important to us anyway. It simply shows us how an error is shown.

```
SALEM>KB7NZ <I C S3 R2>:Hello Bob, Welcome to N7IFJ's MSYS
BBS in Salem, OR

SALEM>KB7NZ <I C S4 R2>:Use  D HELP.DOC to download a help
file (Tnx to KA7ZIN),
 enjoy.

SALEM>KB7NZ <I C S5 R2>:Enter command:
A,B,C,D,G,H,I,J,K,L,M,N,P,R,S,T,U,V,W,X,?,* >
```

The preceding example does not really give a sense of the activity during the monitoring interval. This was observed over an interval of only a few minutes. And there were only three users present. Yet there were many packets passed! Among the three users, there were remarkably few retries. This is a very desirable condition but not that common when things are busy.

4.E MORE ABOUT SSIDs

There have already been 3 discussions concerning SSIDs (step 8 of "Setting Up the Modem" in section 2.E.1, section 3.A.1 and section 3.B). Here, the focus is on what happens to an SSID in a node.

First, consider the SSID for your own station. When you set the TNC2 parameter *mycall callsign* and just use your own call, a SSID of "0" is assumed. Other SSIDs for TNC2s is set with *mycall callsign-SSID*. For example, if N9AA wants an SSID of 5 for a TNC2, the entry would be *mycall N9AA-5*.

When you use a Net/Rom-style networking node (which is the case throughout the west), the node adopts your call (modified) for all outgoing packets to anywhere other than your station. It puts an SSID on the call. The number it uses is 15-(your SSID). Thus, a node to which KA7EHK connects becomes to others KA7EH-15. KaNodes subtract 1 (MOD 16) from the SSID; an SSID coming in as 5 comes out 4 and one coming in as 0 goes out 15.

But why all the monkey business with SSIDs? Suppose that you connect to your friendly local node and tell it to connect you to W0XYZ. Suppose, furthermore, that W0XYZ is not very far from you. The connect request is successful. What would happen if that SSID is not changed? Your TNC could not tell whether W0XYZ's packets are for the node or for you, directly. Since the node is (usually) your call with an SSID of 15, returning packets are addressed to that SSID and there is no confusion.

There is a potential trap involving combinations of Net/Rom nodes and KaNodes. It has to do with the differing ways the two change incoming SSIDs into outgoing SSIDs. The author discovered this situation while exploring a local KANode. Suppose that you adopt an SSID of 0 and connect to a local KaNode. Then, on the SAME frequency, you connect to a Net/Rom-style node. The outgoing SSID from the KANode is 0-1=15 and that is the incoming SSID at the Net/Rom node. If you then try to connect to another local station on the same port of the Net/Rom node as that on which you entered, you go out as SSID = 0. Now, the KANode can't distinguish between your packets and those from the Net/Rom node and a fearful lockup may ensue. There are several other combinations where the same result may occur.

It is also worthwhile noting about SSIDs as packets go through a network of Net/Rom. Any time you autoroute to a neighboring node, you appear with the same SSID you would outgoing locally. It makes no difference how many nodes you pass through; the SSID stays the same. When you make an outgoing connect at a distant node, it is with the same SSID as it would be with a local connect.

In section 4.D.2, it was mentioned that hearing a station with an SSID of 15 but not hearing the call without an SSID is an indicator of a hidden transmitter. This is true. But there is also another reason for this to happen. If a station at a distant node has made a network connection to your node, then made a connection out to a local station, you will also see only the call with an SSID. The only way to tell which is the case is to connect to the node and obtain a User list (see Chapter 24 in Volume 2 for more information).

4.F OPERATING ON NODE FREQUENCIES

After antennas, the next most common point of contention is about operating on the same frequency as an active node without using that node. As with most such debates, there are of course, at least two correct answers!

4.F.1 ADVANTAGES OF OPERATING WITHOUT A NODE: If you do connect direct to another packet station (say a BBS) on a node frequency, you reduce the number of packets by 50%. Instead of one packet between you and the node and a second packet between the node and the other station, there is only one. Thus, your impact on the frequency is only half.

An observation is in order: if you do choose to operate this way and it is possible for both stations to change frequencies, do so!. If 145.01MHz is a no-node frequency in your area, use it. This also appears to be a legitimate use of 145.50-145.80MHz for this is genuinely miscellaneous (this is an FCC set-aside no-repeater subband). Likewise, 146.40-146.58MHz is simplex (ARRL Band Plan) and packet is, without any debate, simplex. Any of these frequencies away from normal packet operation could not help but go better than trying to compete with a node and its users!

4.F.2 DISADVANTAGES OF OPERATING WITHOUT A NODE: One of the arguments against direct connects on a node frequency is that the node serves as traffic control on the frequency. Lets look at this argument and attempt to determine whether it is valid.

Suppose that that there is a hilltop node called NODE. Two users, A & B, are on one side of the hill. Another user, C, is on the same side of the hill as A & B. C is talking to D though NODE while A is talking to B directly. So, what happens? A,B,C, and NODE all hear each other; NODE and D hear each other. A, B, & C avoid collisions simply because they can sense each other's carriers. This happens without regard to the presence of NODE. D, on the other hand, collides with A,B, & C as heard by NODE (as well as the other way around) because they are hidden from each other. NODE is in a different situation; when it has a packet to send from D to C (or C to D), it waits its turn because it can hear all four.

Now, suppose that A & B change their link so that NODE is used. Is it any better? The number of packets increases as pointed out in section 4.F.1. The RR (acknowledgement) packets issued by NODE appears to modify the transmit pattern a little and reduce the collisions heard by NODE. But the effect does not seem to be large.

Comment: it does appear that a small improvement in collision rate at the node does happen. There certainly are more packets on the frequency. If the collision rate is, indeed, reduced, there will be fewer retries but, again, the reduction seems small. Thus, it appears that the cost of operating on a node frequency without the node is not very high. But, why battle? Change frequency if possible!

4.G TNC PARAMETERS

TNCs have basically three sets of parameters. One set controls the radio operation of the TNC. One set controls the relationship between your terminal and the TNC. The last set controls the convenience features.

In the following discussion based on TNC2 style commands, there is usuall a short and long version of each command. The short version is usually the first few letters of the long version. The short vesion may be used by itself; if the long version is used, it must be spelled correctly and in full. Most TNCs accept either upper or lower case letters in the commands. Commands will be denoted as XX[yyy] where XX is the required short form of the command and yyy is the optional remainder of the command.

4.G.1 RADIO PARAMETERS: In TNC2 style, these are AXD[elay], AXH[ang], CH[eck], DW[ait], FR[ack], MAX[frame], MY[call], P[aclen], RES[ptime], RE[try] & TX[delay]. With some newer TNCs which use somewhat different internal rules for timing retries, SLOT[time] and PER[sist] are also in this group of parameters.

Unless you are using a voice repeater, AXD[elay] and AXH[ang] should be 0. AXD[elay] accounts for slow response time of most voice repeaters and AXH[ang] accounts for the usual voice repeater squelch tail.

DW[ait] is used to account for characteristics of digipeaters but affects general operation also. For TNC2s the value of DW[ait] is in 10mS (10 milliseconds or .01 seconds). A value of 1 produces a 10mS wait; a value of 15 produces a 150mS wait. This wait is between when the TNC detects a clear frequency and when it begins to transmit.

TXD[elay] is used to control two aspects of operation. It, also, is a time interval. Also, in TNC2s, it is in units of 10mS. On every transmission, when the transmitter is keyed, there is a wait until data sending actually begins. This is because some transmitters take much longer than others to establish normal operation when keyed.   This parameter is also used to reduce collisions on retries. On retries, an extra delay is added to DW[ait]. This delay is a random amount. The idea is that if two transmissions collide, if they wait exactly the same interval for their retries, they will collide again. The extra delay is a random number times TXD[elay]; the random number varies on each retry from 0 to 15. Thus, after the first (unsuccessful) try, the wait might be DW[ait] + 7*TXD[elay]. The next time, it might be DW[ait] + 3*TXD[elay]. Remember, this is the interval between sensing a clear frequency and transmitter keyup. There is still always a wait of TXD[elay] between transmitter keyup and start of data.

FR[ack] controls how long the TNC waits for an acknowledgement before assuming that none is coming. In TNC2s, this is in units of 1 second. When a digipeater is used, the wait time is lengthened according to the number of digis used (at the rate of 2 FR[ack] intervals per digi). If this value is set too small, your TNC will send retry packets when they are not needed. FR[ack] should generally be set quite a bit longer than RES[ptime].

RES[ptime] is the complement of FR[ack]. This is the MINIMUM wait time for packet acknowledgements. For TNC2s, this is in units of 100mS (0.1 seconds).   It comes into play primarily when many packets are being sent rapidly as in file transfers.

P[aclen] sets the largest packet which your TNC will send. It has no effect on received packets. This value normally comes into effect only when you do not do enter/return at the right edge of the screen!

MAX[frame] sets the number of unsent packets which are allowed to build up in your TNC. When this number is reached, the TNC does not accept any more characters from your terminal until there is room (if Flow is ON, the TNC2 default). MAX[frame] also sets the maximum number of packets which may be send it a continuous block.

RE[try] sets the number of times your TNC will retry to send a packet before it gives up.   Once this number of retries have been attempted, a disconnect request is sent. If you have a good path to the station you are talking with, this number can be fairly small (say 4). If the path is poor or the frequency is very busy, it may need increasing. Be aware, however, as your retries increase, it increases congestion and may cause others to increase their RE]try] value! The effect of this death-spiral should be apparent.

CH]eck] sets the connection time-out interval. For TNC2,s the units are 10 seconds. The default value for TNC2 is 30 for 300 seconds or 5 minutes. If there has been no activity for this interval, your TNC sends a query packet to the other station. It retries according to the value for RE[try]. If no response is heard, a disconnect is sent.

MY]call] sets the callsign of your TNC. It is illegal to operate without your call entered into the TNC! The default value is NOCALL for TNC2s. Most nodes and other TNCs will not accept connect requests from NOCALL. If you suddenly find your connect requests being rejected, check your callsign!

Check Chapter 3, section 3.F.7, for a discussion of slot time and persistance.

It is easier to describe these parameters than to advise on proper settings. TXD]elay] depends primarily on the characteristics of your transmitter. FR[ack], DW[ait], and P[aclen] should be similar for all stations in your area.

4.G.2 TERMINAL PARAMETERS: These parameters set the interaction between the TNC and your terminal. There is a lot of variation in these particular parameters from manufacturer to manufacturer. Fortunately, for most terminal programs, you do not have to change much because the defaults fit pretty well. Furthermore, most terminal programs are now flexible enough so that you can change the terminal configuration to fit the TNC rather than the other way around.

Things you might particularly want to check if you are having problems are AUtolf, AWlen (sets number of data bits in characters), Echo, & PARity.

It is the author's strong opinion that you should use the TNC's default terminal parameters and make the appropriate accomodation in the terminal. See section 4.M for the reason why.

4.G.3 CONVENIENCE PARAMETERS: These include a great variety of things from monitor status to beacons.

Some advice may be appropriate here. If you have a TNC with CWID (morse code identification), TURN IT OFF! The need for CWID has long passed.

Beacons are another touchy subject. If you feel you must beacon, do it as infrequently as you can. For TNC2s, the beacon interval is in units of 10 seconds up to 250 (2500 seconds). 2500 seconds is just over 40 minutes. This interval is not too short! Also, if you must beacon, keep the text short. Otherwise, it just clutters the frequency.

A pair of convenience parameters which the author likes are the TNC2 NEW[mode] and NO[mode]. If NO[mode] is off, the switching between command mode and converse mode is controlled by NEW[mode]; if it is on, switching between between command and converse modes happens only by direct command. I prefer NO[mode] off. NEW[mode] off causes the switch to converse mode from command mode to happen automatically every time a new connect occurs. Also, with NO[mode] off, the TNC stays in converse mode when a disconnect occurs. But with NEW[mode] on, the switch is automatic going both ways.

The U[sers] command (TNC2) controls how many incoming connections are accepted by your TNC. Until you get some practice with handling several conversations at once, set this value to 1. This way, any multiple conversations must be initiated by you.

4.H MAIL BOXES

Several previous discussions have shown mailbox operation (see sections 3.B and 4.D.2). Here the focus will be on how your own mailbox operates (if you have one) and how to create one of you do not have one.

4.H.1 TNC MAIL BOXES: TNC mailboxes reside within your TNC. They stay operating when your computer is turned off or used for non-packet activities. As previously mentioned concerning SSIDs (section 3.B), there are generally two classes of TNC mailboxes. One is the full-time mailbox and usually has a specific SSID assigned to it. The part-time mailbox uses the same SSID as you use but the mailbox cannot function when you are conversing through the TNC.

With the exception of several newer TNCs with mail-snatch capability (that is, they can automatically connect to your local BBS and retrieve your mail), these TNCs take very little to set them up. The full-time mailboxes need an SSID assigned. You may also be able to set the connect message. In the case of part-time mail-boxes, you must remember to turn it on and off at the appropriate times. The author turns his on when packet operation ends and turns it back off (after checking for new mail) when packet operation resumes.

Some TNCs have specific requirements to permit a mailbox to operate. For example, MFJ 1270/1274 TNCs require *user 1* and the streamswitch set to stream A.

4.H.2 SOFTWARE MAIL BOXES: software mailboxes reside in the terminal program. These range from NET/NOS at the high-end (see Chapter 14 and, in Volume 2, Chapter 25) to the popular LanLink (see Chapter 18 in Volume 2). Mailboxes for BayCom and DigiCom modems fall in this category, also.

Software mailboxes are usually controlled by a script or config file. The writing of such a file is beyond the scope of this book. If the documentation is not adequate, you will have to rely on experience of other users.

4.H.3 MAKING A MAILBOX: Ok, so you don't have a TNC with a built-in mailbox and you don't want to tie up a computer with one of the terminal programs which has a mailbox in it. What can be done to get the effect of a mailbox? Fortunately, its pretty easy.

The strategy is this: there are flow-control commands which tell your TNC when it can send characters to the terminal. If you tell the TNC to stop sending, it simply stores characters in it memory until you tell it to begin again. This works so long as there are fewer characters to be stored than the capacity of the TNC memory! So, how do you use this?

First, look up in the TNC manual under *XON* and *XOFF*. The character specified by *XOFF* tells the TNC to stop sending. The character specified by *XON* tells the TNC to start sending. For TNC2s, the defaults are XOFF = $13 (control-S) and XON= $11 (control-Q). Assuming that your TNC responds to TNC2 defaults, try the following experiment:

(1) Send XOFF character (control-S) to your TNC. No carriage returns are needed; just send the single character.

(2) Press the enter or return key 3 times. You should not see anything happen on the screen.

(3) Send XON character (control-Q) to your TNC. No carriage returns are needed; just send the single character.

(4) Three command prompts should appear on the screen after step 3. This is because those characters were stored in the TNC and were not released until you told the TNC that it is ok.

If this experiment works, you have a rudimentary mailbox! To use it, compose a CTEXT to be used when you are not present. This is a message which is sent to anybody who connects when CMSG is ON. If CMSG is OFF, no CTEXT is sent to a connecting station. The author has used the following:

CTEXT KA7EHK message buffer is open for your use; please leave msg

When you end packet operation, simply send the following commands to your TNC:

```
cmd: CMSG ON
cmd:  (xoff character)
```

Then, when you resume packet operation, send the following. Note that there are no command prompts (cmd:) because they have been held inside the TNC!

```
CMSG OFF
 (xon character)
```

Anything which the TNC received between the times XOFF was sent and XON was sent will come spilling out onto your computer screen! You now have a very simple mail box! You do need also to remember to leave your radio and TNC on; you computer may be off or devoted to other uses.

4.I THE TNC LIGHT SHOW

Most stand-alone TNCs (with the exception of BayCom and DigiCom which really aren't TNCs) have a number of status lights on their front panel. These lights can provide some very useful indications of the activity of your packet station and of the frequency your radio is tuned to.

4.I.1 POWER: the PWR indicator, if you have one, normally just sits there and glows. But if you are having problems, it is one of the first places to look. If it is not on, then power is missing. Check to see that the power cable for the TNC is still in place and that the main power supply or 115v connect is good. If these check out, then there may be more serious problems.

4.I.2 CON: the CON light, if you have one, indicates that your TNC is connected to another station. It should go on as soon as you see a connected message on the screen; it should go out as soon as you see the disconnected message.

The is a fairly significant but not obvious trap here. If you have a TNC2-clone (or similar), with multiple connect streams, the CON light shows the connect status of the stream you have currently selected. If there is no connection on that stream, the CON light will go out even though there are connects on other streams.

TNC2s have a command called connect-status (*CS*) which lists the connection status of all the possible connection streams. It also shows which stream is the current one. This helps to resolve how many connections you have and who they are with.

4.I.3 STA: the STA light indicates the status of outgoing packets. It stays lit as long as there is a packet waiting to be sent or one which has been sent but not acknowledged. Again, this light shows the status of the currently selected stream; even though all packets may have been delivered on other connections, it will stay lit if your current stream has an undelivered packet. Likewise, if all packets have been delivered on the current stream, it will go out even if there are undelivered packets on other streams.

4.1.4 PTT: the PTT light (Push To Talk) goes on each time your transmitter is keyed. This is a useful indicator of proper connections between your TNC and transmitter. If the PTT light flashes but there is no transmit indication on the transmitter, there is a problem with the radio-to-TNC cable or connectors.

The PTT light is an indicator which shows when your station is being used as a digipeater. If no connect light is on but the PTT light continues to flash, then your station is being used! It may also flash during a connect request before the actual connect is successful.

4.1.5 DCD: the DCD (Data Carrier Detect) light is very useful. It lights when packets are detected on your frequency. It lights when ANY packets are detected, not just addressed to you. Thus, if it is flashing frequently, the frequency is quite busy.

4.I.6 OTHER TNCs: Other TNCs may have different indicators than the ones discussed above. In particular, the Data Engine has a set which depends entirely on the installed software. They do not necessarily behave the same with JNOS40, G8BPQ, or other software.

Look at the owner's manual for your TNC because these lights can tell you a lot about how your station is operating. It should also be noted that many of the multi-mode TNCs have many more indicators than discussed here; some show the current mode. Familiarise yourself with these because they can often tell you whats wrong when things are not right.

4.J. The TNC Internal Clock

The internal clock used by TNC2s for time marking (time stamping) has an interesting quirk. The folk-tale goes something like this: The TNC was designed around a particular clock crystal frequency. Baud rates were determined according to this frequency as well as the time clock. The software was all written and units built. It was only then that somebody noticed that a harmonic of the crystal frequency fell so close to one of the standard packet frequencies that it was nearly unusable. The fix for this problem was to tune it down slightly in frequency. There would be no real problem with baud rates because the change needed was quite small. But the change resulted in a clock which looses a few seconds (or more) each day. Thus, standard TNC2 clones need constant resetting of the time.

4.K. Computer-TNC Baud Rates

The real throughput of a TNC depends on the radio baud rate (and other things like DWAIT and retries). The on-screen display certainly looks nicer if there is a faster response between the TNC and the computer screen.

TNC2s permit the computer baud rate to be set as high as 9600 baud. Most users who try this rate discover that characters are lost or garbled. The problem is the integrated circuit which is used to convert between computer logic levels and the voltages required for RS-232. It is an operational amplifier (op-amp) and the one used operates rather slowly. The specific integrated circuit is called a quad op-amp (that is, it contains four op-amps) and is an LM324.

This op-amp may be replaced with a much faster TL084. TL084s are available at many hobby electronics shops where components are sold. Simply unplug the existing device and plug the new one in its place. Note the notch on one end of the original and put in the new one with the same orientation. In a standard TNC2, with the front toward you, this circuit is near the right side of the circuit board about mid-way back.

4.L. Station Layout

The actual layout of a station is more important than one might realize. This is because there are at least three devices in a packet station which can interfere with each other. The transmitter can interfere with TNC and computer operation. The TNC can interfere with receiver operation (as mentioned in section 4.J). The computer can interfere with receiver operation.

Most of these problems are simplified if the station antenna is well separated from the station (30' or 10 meters is not too little). The worst situation is when you must use a small antenna very close to the TNC and computer. Transmitters can cause computers or TNCs to malfunction, and can cause video displays to wiggle. The author has seen and experienced many problems with antennas as close at 10' or 3 meters to computers and TNCs. This is true whether operating on HF or VHF.

When the antenna must be close to your other equipment (such as an emergency station or a portable one), careful choice of the computer is a must. Flat panel computer displays such as those used with laptop computers tend to be less affected by trans-mitters. Also, some computers are more quiet than others; experience of other hams may be an important guide. You may also need to add filtering capacitors at the TNC on the connector for the TNC-computer cable. Be particularly careful that all metal parts of the transceiver enclosure are connected together electrically. Special metal boxes for transceivers may also be necessary.

4.M. When a TNC Quits

Sooner or later, almost every packet operator comes face-to-face with a TNC which does not respond as it usually does. The author has had to deal with this situation several times. Here are some things which may help.

4.M.1 Why IT Happens: It is actually rather difficult to say in every case why a TNC stops operating. Several of the author's cases seem to be associated with very fast power line drops and recoveries. Others have been quite mysterious.

4.M.2 What Happens: One common effect is that the serial computer link settings get changed in the TNC. If the baud rate is set by switch, this seems pretty unlikely to change. But changes in parity, data bit, and stop bit settings have been observed. Other changes seem somewhat deeper than mere serial settings.

4.M.3. What to Do?: The author's first step has usually been to try to re-establish serial communication between the TNC and the terminal.

Step 1: First, try several *return/enter*. One of three things might happen. You might get no response at all. You might get a cmd: (or perhaps a scrambling of these characters). The TNC may try to send packets (as shown by the PTT indicator). If the latter case is true, the TNC is in CONVERSE mode; try a *^C* to see if the command prompt shows. If you get a cmd: then you are close. If there is no response at all, you need to go on to the next step.

If you get a cmd: or something close to this, try asking the TNC what its parity (*PARity*) is; make certain that your terminal settings are the same. Then change the TNC parity back to what you want it to be first and second, change the terminal program to the same value. Do the same for number of data bits (*AWLEN*). If neither of these work, go on to the next step.

Step 2: Sometimes, what appears to be TNC failure is caused by the TNC being in an unexpected mode. These include KISS or HOST (the latter primarily in AEA or Kantronics TNCs). If your TNC is in either of these modes, it is unlikely that you you will be able to restore operation without resorting to the last step. But you cannot know whether or not this is true, so continue!

Step 3: Another reason for unexpected TNC operation in multi-mode TNCs is caused by the TNC being placed into a communications mode (RTTY, CW, AMTOR, etc) which is different than the one you expect. If you have a multi-mode TNC, check with your manual about switching between communication modes. If your TNC is not multi-mode or if you are not successful, continue to the next step.

Step 4: Try typing *RESTART* even if you get no other response on the screen. Returns the TNC to the same condition as if it were being turned on. If the faulty operation is due to a parameter (which is normally stored in the battery-backed memory) being changed, you may see no improvement. But, then again, sometimes it works.

Step 5: Try turning off your TNC, then turn it back on. Again, there may be several responses. You may get the normal start-up message and all will be more-or-less normal. If this is the case, you may need to reset *MYCALL* and any other changes from default operation. The author keeps a card which lists these changes in the front of the TNC manual.

You may get the normal startup message with some characters scrambled. In this case, the serial settings of the terminal are different than that of the TNC. Try changing these settings carefully, one at a time, until the message appears normal, then check to see whether other settings need to be changed.

You may not get any response at all. If this is the case, go to the next step.

Step 6: If you get no recognizable response when you turn on your TNC, something is improperly stored in the TNC's memory. The memory needs to be cleared of all of its old information so it can be restarted as if new. In a normally operating TNC, the *RESET* command will do this. Try typing out *RESET* because if the TNC recognizes it, it will save doing the next step. If it is successful, you will see the standard startup message. If it is not successful, proceed to the next step.

Step 7: To clear the TNC memory, you need to understand that the TNC normally has a small battery in it to provide power for the memory when external power is turned off. You need to disrupt the battery power for a few minutes in order to clear the memory. Some TNCs (such as AEA PK-232s) have a jumper which forms a simple switch; this jumper can be removed to disrupt battery power; check with your manual to determine whether or not your TNC comes this way. Other TNCs have no jumper (as in MFJ 1270/1274). In this case, you need to break the circuit from the battery. These TNCs usually have a simple holder for the battery; you can slide a piece of paper under one of the battery spring clips to break the circuit.

When the battery is disconnected, turn off the TNC and wait several minutes. Reconnect the battery and turn on the TNC. Normal operation should be restored. You will have to reset *MYCALL* and any other non-default TNC settings.