Here, a local conversation is one not using a node. Please note: in many areas, it is not considered "good practice" to carry on a conversation on a frequency with a local node without using that node; the reason for this should become clearer in Chapter 4. This section will show you how to make a connection without using a node. Understand that this may just be a learning step, depending on your local practice.

The following typographic conventions will be used in this and following sections: text FROM another station (including a node or BBS) are in this font; text from your TNC is in this font; *text you type is italicized.* (note that it does NOT appear italicized on your screen!)

A very useful metaphor for various aspects of packet radio is the telephone system. This will be used over and over to try to make the concepts somewhat more intuitive.

3.A MAKING A CONNECTION

Making a direct connection is like calling someone up on an old-style party line. You didn't need to use the operator. You could call the Smiths down the road by ringing 2 long rings and 1 short ring on the hand-crank phone.

You can do the same on packet. If you want to talk to W9ABC, you tell your TNC: "connect me to W9ABC". The TNC constructs an appropriate packet with your call, the call of the station you want to connect to, and the request to connect. The TNC causes this message to be sent out on your radio and it listens for a response from the other station. If there is no response, it tries again; this is called a "retry". The TNC tries a number of times controlled by a value which you can set in the TNC, then gives up if there is no answer.

3.A.1 FINDING SOMEONE TO TALK TO: One of the big questions asked by almost every new-comer to packet is: "How do I find somebody to talk to?" This question needs to be answered before discussing how to make your first connection. The way to start out is to use the "heard" list kept by your TNC. In TNC2 clones, you get this list in the following way:

```
cmd: mh
AF7S-1       24-02-93  15:04:45
KB7QQH       24-02-93  15:04:18
SALEM        24-02-93  15:04:17
AA7JI        24-02-93  14:57:42
WA7SMO-15    24-02-93  14:54:39
K7IQI        24-02-93  14:53:13
K7MYU        24-02-93  14:51:39
N7ICS        24-02-93  14:41:43
K7UN         24-02-93  14:36:49
WA6SDR       24-02-93  14:13:09
cmd:
```

Note that this list comes from your TNC. Don't worry if yours does not have time and date or looks a little different. Note that some are not normal calls but names, like "SALEM" in the list above. These are usually from nodes or bulletin boards. Others are more-or-less "normal" ham calls. A few others look like ham calls with numbers on the end. These numbers have been referred to before as "SSID"s (Secondary Station ID). Many SSIDs indicate a node or a packet which has passed through a node; this will be discussed in more detail in later chapters. Others indicate mailboxes. Those callsigns without SSID's are prime candidates to talk to.

3.A.2 MAKING A CONNECTION: Lets try to connect to a station which isn't on the list. To request a connection, after the command prompt (remember that this means that you are "talking" to the TNC), simply use a *C* (for connect), a space, and the callsign, like this:

```
cmd: c w7abc
```

The TNC and radio will try several times. If the station really isn't there, the TNC responds with:

```
cmd:*** retry count exceeded
*** DISCONNECTED
```

This message tells you that the TNC tried in vain. Here is what happens when a connect is tried to a station which IS on the list:

```
cmd: c kb7hgj
cmd:*** CONNECTED to KB7HGJ
Please leave message @ KB7HGJ-1 Thanks <de Kevin> Chehalem
Mtn. Newberg,Or.
Hello, anybody home?
cmd: d
cmd:*** DISCONNECTED
```

Lets walk through what just happened. The connect command was given in the 1st line. The TNC responds with the *** CONNECTED to KB7HGJ message in line 2. This message comes from the TNC, not the other station. The other station, in this case, responds with a "connect message" in lines 3 & 4; this tells you to connect to KB7HGJ-1 in order to leave a message (see mailboxes in the next section). The typed query in line 5 goes unanswered, so a disconnect was made in line 6. This does not really show how the disconnect happened. In these simple cases, you must tell you TNC to disconnect you. Thus, YOU have to place the TNC into command mode; for most TNCs, a control-C (no ENTER or RETURN is needed) does it. The *d* is short-hand for "disconnect" and the TNC informs you when the task is finally accomplished.

Now, if someone actually had been on the other end, you could type back and forth. Once the connection occurs, the TNC automatically switches into "converse" or conversation mode. In this mode, the TNC is rather transparent, just sending on through (within some limits) what you type. Similarly, it receives what was typed on the other end and presents it to your terminal for display.

3.B MAILBOXES

Many TNC's have built in mail-boxes. If you leave your radio and TNC turned on all the time, other packet stations can leave mail in your TNC.

Mail-boxes are much like simple bulletin boards so we won't go into great detail now. There are several styles. A common one looks like this when you connect:

```
[KAM-5.0-HM$]
6218 BYTES AVAILABLE
Welcome to Oregon, please leave me a message.
ENTER COMMAND:  B,J,K,L,R,S, or Help >
H
B(ye)         PBBS WILL DISCONNECT
J(heard)      CALLSIGNS WITH DAYSTAMP
J S(hort)     HEARD CALLSIGNS ONLY
J L(ong)      CALLSIGNS WITH DAYSTAMP AND VIAS
L(ist) [;]    LIST MESSAGES YOU CAN READ
L <|> call    LIST MESSAGES FROM OR TO CALL
LB            LIST BULLETINS
LC [cat]      LIST CATEGORIES
LL n          LIST LAST n MESSAGES
LM(ine)       LIST UNREAD MESSAGES ADDRESSED TO YOU
LT            LIST TRAFFIC
K(ill) n      DELETE MESSAGE NUMBER n
KM(ine)       DELETE ALL READ MESSAGES ADDRESSED TO YOU
R(ead) n      DISPLAY MESSAGE NUMBER n
RM(ine)       READ ALL MESSAGES ADDRESSED TO YOU
S(end) call   SEND MESSAGE TO callsign
SB call       SEND BULLETIN
SP call       SEND PRIVATE
ST call       SEND TRAFFIC
ENTER COMMAND:  B,J,K,L,R,S, or Help >
```

While this one may be a little more elaborate than some, it is not much different than many of the mailboxes on the air. You can disconnect by sending it a *b* (for bye) or you may disconnect from your end.

There is one important distinction between mailboxes; some are full-time and some must be turned on and off. Full-time ones usually use a specific SSID for the mailbox and another SSID for "talk" with the operator (as in the example of the previous section). Part-time mailboxes use only one SSID; these must be turned on when keyboard operation is completed, then turned off when keyboard operation is restarted.

3.C DIGIPEATING

What do you do when the station you wish to talk to is not quite in radio range? The first solution developed for packet radio is called digipeating. While still used, it is no longer the only solution. It has its place and its limitations which are discussed in the last section of this chapter.

3.C.1 CONNECTING USING A DIGI: Digipeating uses one or more other stations to relay your message to a distant destination. Lets suppose that you are on one side of a mountain, W7ABC is on the other, and, as in our example in section 3.B, connect requests are not successful. Lets suppose that AA3ZZ is fortuitously located on top of the mountain. See Figure 3-1. We can use AA3ZZ as a digipeater (or, digi for short) by requesting that our TNC try to connect to W7ABC via AA3ZZ. The command to the TNC looks like this:

`cmd:` *c w7abc v AA3ZZ*

There are several things to note from this command. One is that callsigns can be entered in either upper or lower case. You can spell out *via* or use the shorthand, *v* .

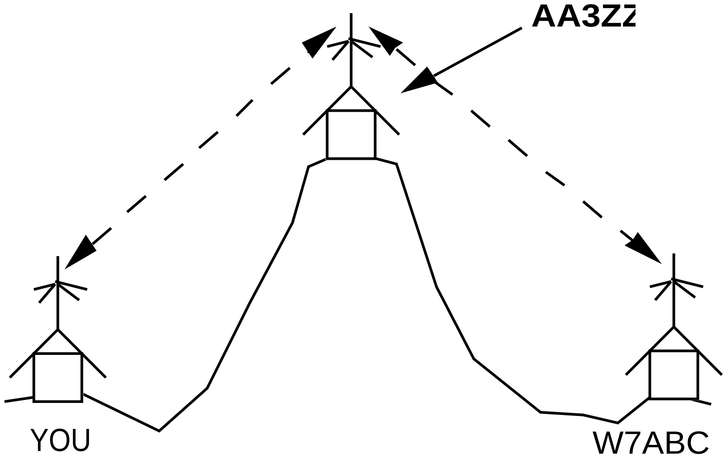

FIGURE 3-1, DIGIPEATI

If more than one digipeater is used (the packet format permits up to 8), the digipeater callsigns are listed in the order in which the packet encounters them leaving your station, separated by commas. Thus, an acceptable request might be

`cmd:` *c w7abc v AA3ZZ, N1A*

where AA3ZZ is the first digipeater encountered by your signal, N1A is the second, and W7ABC, your destination, is the last.

3.C.2 HOW DIGIPEATING WORKS: Every packet has a portion which you do not normally see. This portion, the header, contains the call of the station which sent the packet and the call of destination of the packet. There is also room for a number of digipeater calls.

When a station which is able to digipeat hears its callsign in the digipeat call portion of the header, it listens to the packet. If the packet is received error-free, it resends it. In this way, a whole chain of stations may pass a packet from one to the next.

3.D LIMITATIONS

In the early days of packet radio, digipeating was widely used. One could use a string of digis to go all the way from Portland, Oregon to San Francisco. If this is possible, why is it not as widely used now?

One answer is that digipeaters don't know anything about neighboring digis. You have to find out by asking and exploring. Nodes have solved this problem by managing routing in a much more straight forward manner.

Another answer has to do with error management. Suppose, in our previous example, there was noise which interfered with one of your packets as it was sent by AA3ZZ to W7ABC. If the error wasn't too bad, W7ABC sends back a packet saying that there was an error and your TNC repeats. If the error was really bad, W7ABC's TNC does not even know that there was a packet and simply does not acknowledge (which it does with a successful reception). In this case, also, your TNC waits, then repeats. In both cases, the packet gets resent from you to AA3ZZ and from AA3ZZ to W7ABC. Thus, the repeat packet is resent twice, occupying the frequency twice. If a node were used instead of a digi, the error is resolved between the node and each participant. Repeat packets are only sent between the node and the end station.

Even with these limitations, digipeating still has its place. If you are out of range of a node, digipeating may be an answer. If your local node fails, digipeating may also help (temporarily). Thus, while of more limited use than in the "good old days", digipeating still has a place if used wisely.

3.E ANTENNAS

As mentioned earlier, antennas are a major discussion point. Among packeteers, the issue is usually between directional and non-directional antennas. Here, we will try to look at the advantages and disadvantages of each. In general, the conclusion seems to be that there is no one "right" answer. It may, in fact, take some experimentation to determine what is best for you. Lets first talk about some of the issues before looking at how each antenna type stands up.

3.E.1 MULTI-PATH: Multi-path is a problem which arises when radio waves follow two different routes from the transmitter to the receiver. The end result happens when the two routes are of different length. If the length differences are just right, the two signals can interfere with each other "destructively". Multi-path is quite obvious when watching television; it produces the irritating "ghost" images. It can also be quite obvious when operating VHF mobile; there are spots where signals fade almost completely and there are others, often only a few feet away, where signals are strong.

The effect can be seen on any frequency, including HF. It is easily seen on VHF and often seems more pronounced as you go up in frequency. For a variety of reasons, multi-path problems increase at higher RF baud rates.

3.E.2 HIDDEN TRANSMITTERS: The hidden-transmitter problem is as old as packet radio. This problem is the result of the "carrier sense" scheme which TNCs use to avoid packet collisions. A collision occurs when two packet stations transmit at the same time, spoiling all or part of the other's packet. TNCs try to avoid this by not transmitting when another is transmitting. But if a frequency is vacant for a little while, two TNCs may decide to transmit at about the same time, each unaware of what the other is going to do. This part is an unavoidable consequence of time-shared operation with the protocol (set of rules) which govern our TNCs.

The hidden transmitter problem is in addition to the situation just described. To see its causes and effects, lets go back to section 3.C where digipeating is discussed. In that example, there is a hilltop station, AA3ZZ. It serves as a digipeater for two stations, (you and W7ABC) on opposite sides of the hill. Now, suppose that W7ABC has started to transmit. Your station cannot hear it because it is on the other side of the hill. So, your station can transmit, oblivious of the other station.

AA3ZZ is now in a tough situation because it hears both at the same time. This is the hidden transmitter problem.

3.E.3 NON-DIRECTIONAL ADVANTAGES: Non-directional antennas have been the historical choice for many packet stations. The reasons are fairly clear. With stations in a variety of directions, non-directional antennas are much easier.   Particularly when digipeating is involved, signals may come from several directions at the same time, making non-directional antennas almost a requirement.

Non-directional antennas are fairly inexpensive to purchase when compared to directional antennas. Because of simpler construction, non-directional antennas tend to be more resistant to severe weather but this is not always true.

Non-directional antennas are simple to construct. J-poles and gound-plane verticals are frequently used. 5/8-wave verticals can supply some gain over a 1/4-wave vertical at the expense of a little more complexity.

3.E.4 NON-DIRECTIONAL DISADVANTAGES: One of the major disadvantages of non-directional antennas is the inability to discriminate between signals involved in multi-path problems.

3.E.5 DIRECTIONAL ADVANTAGES: The gain of a directional antenna may provide enough added signal on both receive and transmit to improve operation in low-signal areas. If low signal strength is causing retries, the packet rate from your station (and whom ever you are connected to) will improve. The reduced retry rate will reduce the time your packets occupy the frequency.

If multi-path is a problem at your location, a directional antenna may improve the situation considerably. It is somewhat tricky to aim the antenna for best results, but when you do, the improvement can be considerable.

3.E.6 DIRECTIONAL DISADVANTAGES: A directional antenna makes it difficult for other stations to use yours for a digipeater. In most areas, this is not an issue; but where it is, consider it. If you use nodes in a variety of directions, you may have to re-aim your antenna for each one. This is annoying and you are out the cost of an antenna rotor.

A directional antenna may make it difficult for your station to hear others on the same frequency. Thus, you and those stations become hidden transmitters to each other. This effect may increase problems rather than decrease them.

3.E.7 TESTING: If you have decided to try a directional antenna, it may be worthwhile to borrow one for a short time to see if the results are worth the effort.

One of the best tests is to make a side-by-side comparison with your normal antenna. Set the two antennas up, a short distance apart. Bring coax from both to a point where you can easily change from on to the other. A useful measure is the retry rate. Try connecting to a station or a node (see next chapter). Send a very simple packet (maybe one with just a question mark). Watch how many times the PTT light flashes before the STA light goes out; this light is on when your TNC has an undelivered packet in it. Then change to the other antenna and try the same experiment. You might want to try it several times with each because the result can be effected by the amount of traffic on the frequency. If the retry rate is clearly lower with one antenna as compared to the other, that is likely the "better" one for you.

You might also check to see how the retry rate compares at normal transmit power and the lowest transmit power available. This test tends to emphasize low signal strength conditions.

3.F BASIC TNC PARAMETERS

Most of the TNC parameters are convenience items. There are only a few which effect the performance of your packet station. We will focus on those in this section.

3.F.1 MYCALL: You MUST set your call into the TNC. Usually when a TNC is brand new, the callsign is set to something like "NOCALL". Most nodes and other stations will not accept a connect request from NOCALL. You are also illegal because you are transmitting without identifying yourself! The TNC2 command is: `cmd:`*mycall callsign*.

3.F.2 DIGIPEAT: You can specify whether or not other stations can use yours as a digipeater. It is normally allowed. If there is a reason why you don't want it used this way, the TNC2 command is `cmd:` *digi off* or `cmd:`*digi on.*

3.F.3 DWAIT: This value is important primarily if you use other stations as digipeaters. It helps to keep your transmitter from sending one packet too soon after another which a digipeater may be resending. If you commonly use digis, try increasing this value. In TNC2s, the units you specify are in increments of .01 seconds (that is, 10 milliseconds). You might try setting DWAIT in this case around 0.5 seconds, or `cmd:` *dwait 50.*

3.F.4 PACLEN: The PACLEN parameter controls the maximum length of your transmitted packets. Normally, this only comes into play if you type very long sentences without "returns" or "enters" at the right edge of the screen. When you type a number of characters (including spaces) equal to the PACLEN value, the TNC will automatically send a packet.

You can often improve the retry rate if you make this value smaller when your signal is poor. It is also usually made fairly small on HF. There is a point of diminishing returns, however, because the header of every packet is about 25 characters. If you set PACLEN to 25 characters, the total length is really about 50.

When conditions are very poor, you can achieve the same effect by making your lines short. If you press the enter key about mid-screen (on an 80 character screen), your packet length will be about 40 text characters (for a total length of about 65 characters).

3.F.5 RETRY: This is the number of times your TNC tries to send a packet before it gives up. On a reasonably good frequency, a value of 4 should be adequate. If you keep getting:

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

then your retry value may be too small. This may be happening because the path between you and the other station is poor or because of other activity on the frequency (ie, hidden transmitters!)

But also remember that when you raise the retry value, you are increasing the number of packets on the frequency which makes it harder for others who may increase THEIR retry values which makes it even harder for you, etc.

3.F.6 TXDELAY: This is perhaps one of the most important performance parameters. This value determines how long your transmitter is turned on before the actual packet starts. Older radios often require longer TXDELAY. This is particularly true when the radio has a mechanical transmit/receive relay. If there is such a relay, you can often hear it "click" each time you transmit. For TNC2 type TNCs, the delay units are in steps of 10mS (that is, 0.01 seconds). A value of 1 gives 10ms, a value of 2 gives 20ms, etc. Most slow radios should be adequate with: `cmd:` *TXD 60*. Newer radios should be good with a value of 30 or less.

It is important not to set this value longer than really necessary. A 150 character packet (including header) takes a little over 0.125 seconds. If you make TXD a value of 60, the total time your transmitter is on for each packet is about 0.725 seconds. Thus, most of the "on" time is "warm-up" but it is still occupying the frequency. Thus, this time is really wasting the capacity of the frequency if it is not really necessary.

3.F.7 SLOT TIME & PERSISTENCE: Many newer TNCs have a pair of parameters named slot time and persistence. Once the TNC detects that the frequency is not occupied by another transmitter and it has a packet to send, it waits a time interval of slot time. Then it generates a random number; if the random number is less than the persistence, it transmits; if the random number is greater than the persistence, it does not transmit. If it does not transmit, it waits another slot time and tries the random number again.

In general, the random numbers fall in the range of 0 to 255. Thus, if you set persistence to 128, then, on average, it will transmit on one-half of the tries; if you set persistence to 64, it will transmit on one-quarter of the tries. If you operate on a busy frequency, choose smaller values for persistence; if 4 users is pretty common, try 64 but if 2 or fewer are the rule, then 128 is reasonable. The common rule is to choose a number which is about 256 divided by the average number of simultaneous users on the frequency.

For information about other TNC parameters and how to set them, see section 4.G.