

Building a Packet Network

by Karl Medcalf WK5M

Background

Since the beginning of amateur packet radio, users have tried to push the limits. This has taken many forms: how far can I get, how much data can I pass, how fast can I go.

In 1987, Software 2000, Inc. developed the NET/ROM™ code, which replaced the EPROM in TAPR clone TNCs, in an attempt to improve the packet situation. This code provided the first attempt to build a network using amateur packet radio. Much of the current network system throughout the world is based on this code, and new implementations continue to arrive on the scene.

Purpose

This paper will present various viewpoints on network construction, and does not intend to imply that any one concept is superior to any other. It is intended to provide node operators (current and future) with ideas for consideration to help improve the existing system.

We'll first present the network as it exists in many areas, mostly at 1200 baud on both user ports and backbone ports, and then present ideas for expanding and modifying the existing systems to provide higher throughput and reliability.

Network Concepts

Packet radio networks are intended to provide users with the necessary means to pass data from point A to point B with little regard for the details involved. As such the network node operators must be aware of the impact that any changes and additions to the network may cause.

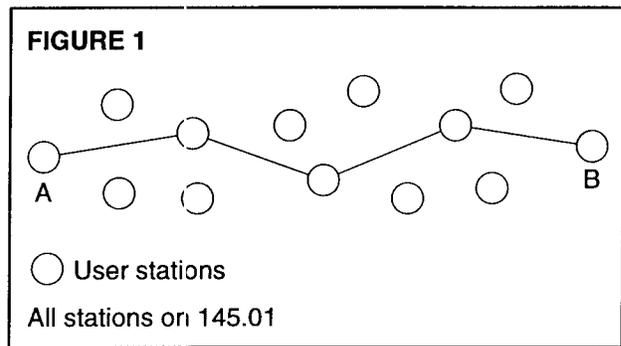
When amateur packet radio began, each packet station was virtually an independent entity. Users could connect to nearby stations and pass data unencumbered, but long distances could not be spanned easily. Every packet station had the ability to be a digipeater, a digital relay station that could retransmit whatever packets it heard, thus

extending the range of communication. In order to use these digipeaters, however, the user had to know the callsign and/or alias of each digipeater from the beginning to the end of the route, and also had to know what order the digipeaters had to be used.

As we look at figure 1, we see how a user might connect to a distant station using these digipeaters. This system, although it was used successfully, had many limitations. In order for this to work, each station in the link had to be able to communicate directly with its neighboring station, and since the AX.25 protocol places a limit of 8 relay stations, this limits the potential range. In addition, each station in the link must be turned on and operating – this could be hit and miss since some people tend to turn their systems off when they aren't operating.

Another defect with this system is that once the link is established through these relay stations, failure of any one station would effectively cause the complete failure of the link. This failure could be due to natural causes (lightning strike, power failure) or through human causes (the user turned off his TNC or changed frequencies on his radio).

Although figure 1 shows that there may be more than one path from point A to point B, the packet system has no means to select the "best" or most reliable path, nor is it capable of detecting failures and routing the data around the failure. With the release by Software 2000 of the NET/ROM firmware, a new era of packet radio began.

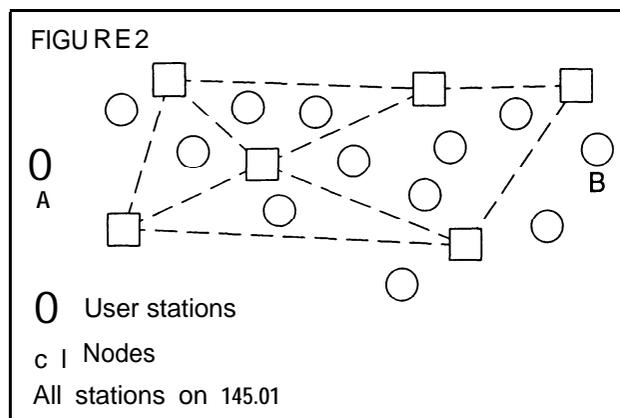


The Beginning of Packet Networks

By replacing the EPROM firmware in the relay stations, these stations became intelligent for routing data from point A to point B. The new firmware converts these TNCs from simple digipeaters into Network "nodes". The nodes broadcast a simple packet on a periodic basis, typically once each hour, and thus other nodes recognize the presence of neighboring nodes automatically. Each node maintains a table of all of the other nodes it can hear (its neighbor nodes), and broadcasts a list of these nodes. Each node would listen to the neighboring node broadcasts, and thus learn of distant nodes - nodes which cannot be heard directly, but which the neighbor can hear. For instance, if Node A can talk to Node B, and Node B can talk to Node C, then Node A can talk to Node C (by relaying through Node B).

The initial network was built using existing **1200** baud TNCs connected to 2-meter radios, and operating on the same frequency that packet operators had been using for some 5 years - 145.01 MHz. This provided a relatively inexpensive means to improve the packet system - no new investment in TNCs, radios, or other hardware was required - only a new EPROM. As we look at figure 2, we see how the network may have looked in the beginning. All stations were operating on a single frequency, and all were operating at **1200** baud.

Problems quickly arose with this system. Since all nodes and all users were on the same frequency, the frequency quickly became overloaded with data. Remember, each node broadcasts its list of other known



nodes periodically and the users were still trying to use the same frequency. Additionally, many nodes were installed at relatively high altitudes enabling them to talk greater distances, but the additional height added to the hidden station problem. A user attempting to connect to a node may suffer collisions from a neighboring node many miles away.

A Better Way

To overcome some of these problems, node operators began building node "stacks" - two or more nodes at the same physical location connected to each other through the serial port of the TNCs. These node stacks would have one node dedicated to "user" access, and the second node, on a different frequency, for node-to-node or "backbone" communication. This scheme allowed the data to pass with little contention over the backbone, and also reduced the hidden station problem since local users were no longer sharing the frequency with distant nodes. As more users and more nodes began to appear, node operators extended the node stacks with additional TNCs. The additional nodes were on different frequencies or bands, and in some cases operating at higher speeds.

High speed backbone systems are seen as one of the keys to success of a network system. Unfortunately, at that time, 9600 baud packet radio was in its infancy, and there were no commercially available radios capable of operating these speeds. Node operators, in their quest for higher speeds, modified existing radios to permit the higher speeds. Today, 9600 baud is rapidly being introduced into networks - partly because the current system is severely overloaded, and partly because the major radio manufacturers now produce radios capable of operating 9600 baud without modification.

One Network Plan

Figure 3 shows one possible plan for developing a network. Each node in the system consists of at least 2 ports - one for the Local Area Network (LAN) to provide user access, and a second for the backbone. The user access port is a 1200 baud port, while the backbone port typically runs at higher speeds -

FIGURE 3

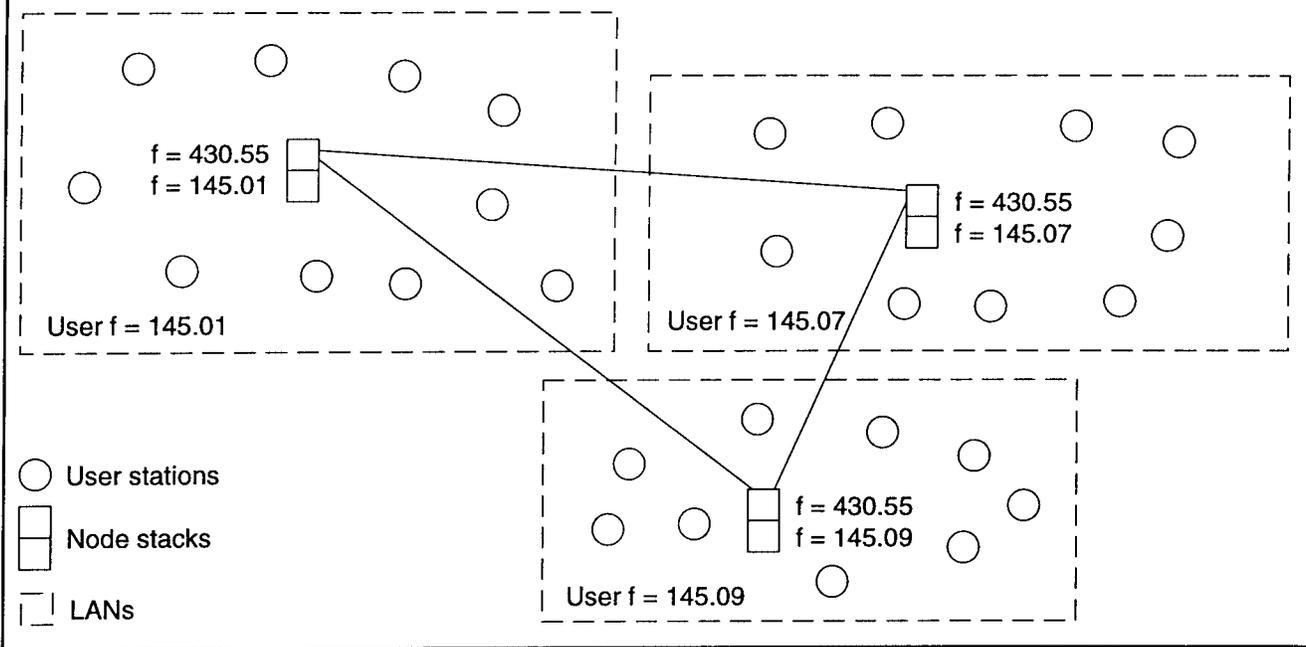
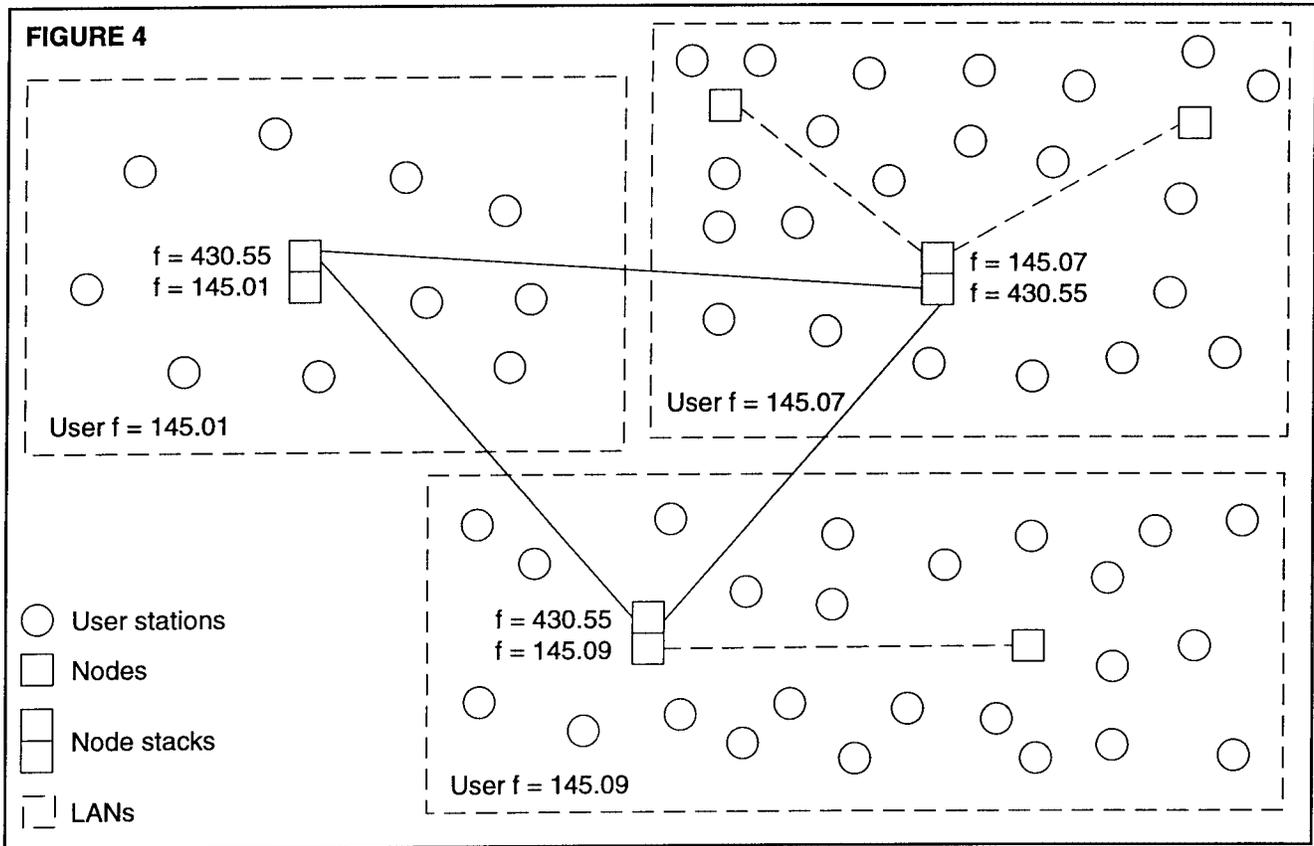


FIGURE 4



9600 baud or higher. Adjacent area LANs operate on different frequencies so they do not interfere with each other. The two ports are connected together at the node to allow users from one LAN to connect to users in another LAN through the backbone.

In this example, the user access port should have its antenna at a relatively low height, perhaps 20 or 30 feet, so that all users in the LAN can reliably connect to the node, but also so that all users in the LAN are close enough that they can hear each other. This reduces the possibility of collisions between users accessing the system. The backbone port antenna is generally located at a fairly high spot, providing the ability to connect to very distant nodes with few intermediate nodes required.

Frequently, users in one LAN may wish to talk to users within the same LAN, and therefore do not require the services of the network. It may well be advisable, therefore, to provide a "direct connect" frequency within the LAN for these purposes. Typically this frequency would not have nodes linked to the backbone system.

In some areas, due to available resources, it may not be feasible to have the LAN cover a small enough area that all users can hear each other. This would normally occur in a lightly populated area where the user base cannot afford to install many dual-port nodes. In these cases, the backbone link may be provided by a single node with two ports, and additional low-speed nodes installed on the LAN frequency to increase the LAN coverage area. Figure 4 shows such a system, expanding figure 3 with additional low-speed nodes.

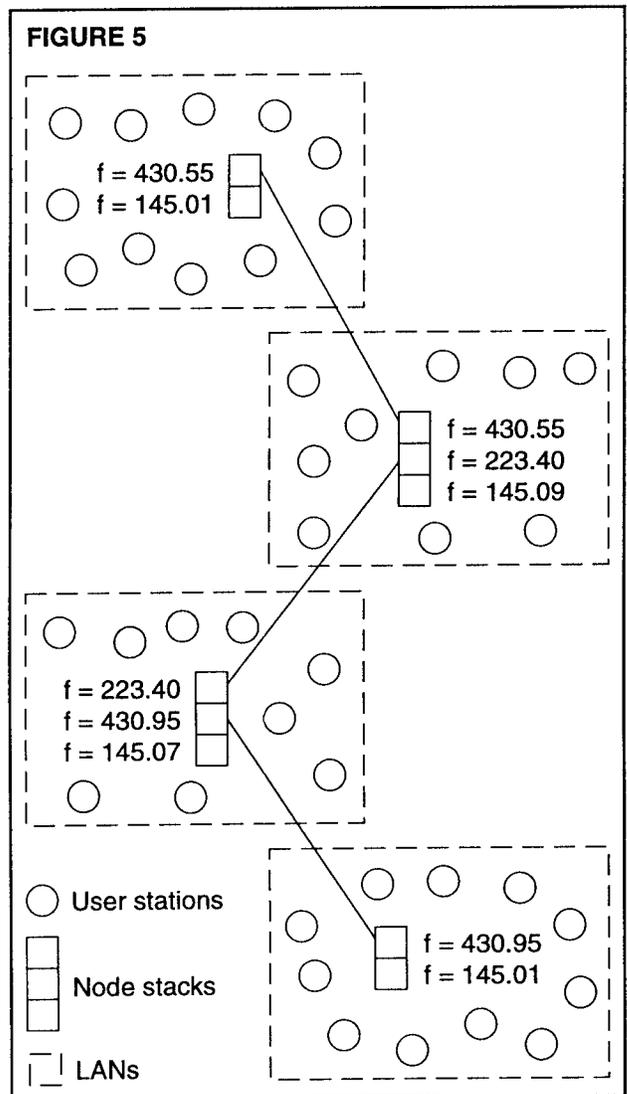
Another possibility

The previous plan can have some drawbacks. With the backbone nodes being capable of hearing very distant nodes, there is the possibility of collisions between the nodes which could severely impact throughput. To overcome this problem, some areas are using staggered frequencies and/or full-duplex between the nodes. Figure 5 shows a possible configuration using this scheme. This scheme

requires that each node be equipped with directional antennas, and must be capable of transmitting and receiving on at least two frequencies for the backbone, plus whatever user frequencies are required. Obviously this can quickly become a cost-limiting factor, but in heavily populated areas the cost may be shared by many users.

Network Loading

Users generally do not present a heavy load to the network system. Since users typically type slowly, and send short messages to the station they are communicating with, the high-speed backbone is very capable of handling several users without problems. However, Bulletin Board Systems (BBSs), DX Packet Cluster Systems, Conference Bridges



and the like present a very different situation to the network.

BBSs provide users with a tremendous amount of information - regardless of what you may think of the value of that information. This data arrives at the BBS from another BBS and so on, which means that the data has to travel through the network at some point. Within a LAN, users access the BBS to retrieve information, but there may be only one BBS in that LAN. Therefore the BBS must use the backbone to send data to other BBSs. This presents a couple of questions to consider: 1) should users access the BBS on the LAN frequency, or should a separate frequency be assigned for user access to the BBS services, and 2) should the BBS forward traffic through the network via the low speed (user) access port of the node, or should the BBS have direct access to the backbone frequency.

There are no concrete answers to these questions, they must be decided on a local basis. Things to consider include the channel loading caused by users accessing the BBS, the cost of the equipment required, available frequencies in the area, and perhaps other "political" factors.

Node Ownership

When networks first started, all that was required was a simple TNC, some new firmware, one radio, and a two-meter antenna. The overall investment to set up a node was relatively small. Those who built the network already had a TNC, and the additional investment for a used radio, small antenna, and cheap power supply was no problem. However, as the network grew, node stacks started appearing, and the cost of multi-port nodes grows quickly. For example, a single two-port node might consist of

1 - 1200 baud packet TNC	\$130
1 - 9600 baud packet TNC	\$200
1 - 2-meter radio	\$300
1 - 2-meter antenna	\$50
1 - 70-cm radio	\$350
1 - 70-cm antenna	\$75
2 - power supplies	\$150
Feed line	\$250

This brings a simple two-port node to a minimum cost around \$1500. Not very many individuals will spend this amount to provide a hilltop node, and if there is a need for a 3- or 4-port node, the cost becomes prohibitive. If you are lucky enough to find a person willing to provide such a node for your system, you're very fortunate, indeed. Consider, however, that private ownership of the nodes comes with potential problems: what if the owner dies, or gets tired of packet, or needs the cash from selling the gear, or moves? You may suddenly find a hole in your network that you can't easily replace.

Perhaps a better solution is for packet groups to sponsor nodes within the area. In such cases, the packet group owns and operates the nodes, so one member of the group moving or tiring of the mode doesn't disrupt the network.

Network Parameters

So now that you've built a network, how does it automatically select the best path from point A to point B? The answer lies in the parameter settings within each node. Network nodes listen for the node broadcast from all neighboring nodes, assign a "quality" to each neighbor, and calculate a quality to each destination the neighbor has reported. These quality figures are then used in the routing of packet data to its final destination.

There are probably as many theories about setting network parameters as there are network node operators. This paper describes one such idea that is being used in the Kansas network and seems to provide reliable node operation.

Quality figures can range from 0 to 255, with 255 being "perfect" and 0 being totally unusable. As we set the quality to a neighbor, there are several items to consider. First, what speed is the link to this neighbor. A 9600 baud link is better than 1200 baud, but 56 KB is even better. If the backbone (link) frequency is on 2-meters and users are also allowed on this frequency, this is not as good as a 2-meter link that is closed to users. Perhaps the link is on 70-cm, which typically has fewer users and therefore should be a higher

quality What is the distance between nodes, and how reliable are the connections between these stations (don't forget to account for varying weather conditions here). A link may be 100% reliable during the winter months when there are no leaves on the trees, and fail miserably when the leaves come out. Rain and high humidity can also affect the reliability of a link.

Given the above, we developed the following guidelines (this is only a portion of the guidelines):

1. A 70-cm link with high-reliability, no users, and operating at 19,200 baud will be assigned a quality of 220.
2. A 70-cm link with high-reliability, no users and operating at 9600 baud is assigned a 200 quality
3. A 2-meter link with high-reliability, users allowed and operating at 1200 baud is assigned a 140 quality

These high-reliability neighbors are "locked in" at the quality shown, and the node is configured so that other nodes heard on these frequencies are assigned 1/2 the quality of the locked in nodes. Figure 6 shows this scheme and the resulting quality for each node.

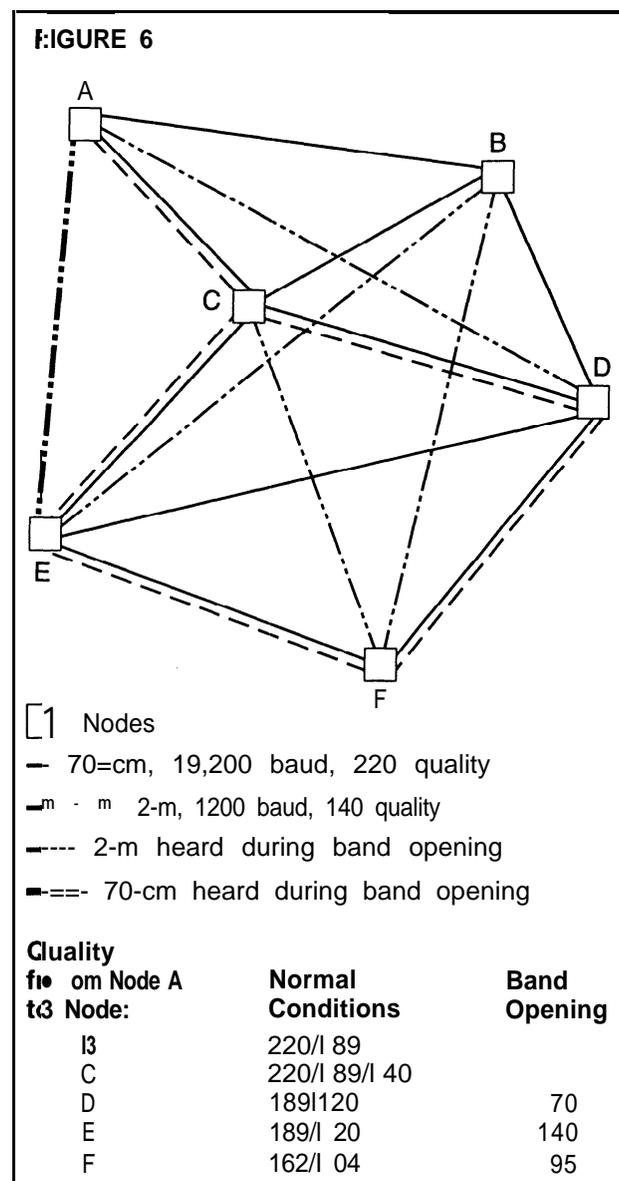
Using this scheme, during band openings, the distant nodes that are heard are assigned a relatively low quality, causing the node to continue to use a more reliable path whenever possible. Although this will tend to limit the number of nodes in the nodes list, a user attempting to connect to one of the distant nodes will probably succeed.

Building A Network

Since the release of NET/ROM in the late 1980's, there have been many derivatives of the networking protocol. The first of these was TheNet, introduced by NORD<>LINK. Other derivatives that followed were the G8BPQ code, developed by John Wiseman, TheNet Plus, various flavors of TheNet XI-J, most TCP/IP NOS programs, and now Kantronics has introduced K-Net™. All of these systems use the same networking

protocol, so your network may consist of a mixture of the various types of nodes. Some of these may offer features not found in others, but the basic networking protocol remains constant.

In the past, installing a node meant that you had to remove the standard user firmware (EPROM) in your TNC and replace it with one that was specifically written for networking. This meant that you gave up the "private" use of your TNC and dedicated it to being a node. In addition, this required you to burn your own EPROM after modifying certain bytes in the image to contain your call-sign, node alias, and your chosen parameters.



To create a two-port node, you needed two TNCs, each equipped with the special EPROM (but each EPROM needed its own callsign and alias). These two TNCs were connected together through their serial ports. As you expanded to a three- or four-port node, more callsigns/aliases were required, and connecting the serial ports together required a somewhat complex diode matrix to avoid data collisions on the serial ports. The number of diodes required is determined by the formula $2*N*(N-1)$ where N is the number of TNCs being connected together. For a three-port node, that's 12 diodes and for a four-port node you need 24 diodes.

With the release of Kantronics K-Net for the KPC-9612, users now have the ability to build a two-port node with only one TNC, providing both a 1200 baud user-access port and a 9600 (or even 19,200) baud backbone port. Since both ports reside in the KPC-9612, the node requires only one callsign and one alias. The unique features of the K-Net in the KPC-9612 don't stop here though - the TNC can still function as an end-user TNC. The PBBS in the KPC-9612 is still usable (there's a BBS command on the K-Net node) and a user can still run multiple connects, a KA-Node™, and even operate Host mode with specialized software.

Configuring a K-Net node is unlike other nodes - all the necessary commands are available at the command prompt. There is no need to patch the EPROM with the callsign and alias as other nodes require. Couple this with the remote access capability and you can place the K-Net/KPC-9612 on a remote hilltop. When the current trustee moves, an authorized user can connect remotely and change the callsign and alias of the node without making a trip to the site. In addition, the K-Net supports the NET/ROM interface over the RS-232 serial port, so it can be easily connected to an existing node stack using other brands of TNCs.

Building a four-port node with two 1200 baud ports and two 9600 baud ports would simply require two KPC9612s with the K-Net firmware, and an interconnecting RS-232 cable with only three wires - ground, TXD, and RXD (TXD and RXD must be crossed). If you

only require one high-speed port but need two 1200 baud user ports, that can be accomplished using a KPC-9612 and a KPC-3 with the optional K-Net firmware EPROMS.

Building or expanding network systems is not extremely difficult, but for the network to operate smoothly and provide maximum benefit, it requires cooperation and coordination between the node operators, BBS system operators, and other network service providers.

DAMA - Another approach

There has been some discussion and previous papers presented on DAMA (Demand Assigned Multiple Access). While this sounds like a completely different networking scheme, it really isn't. The basic premise of DAMA is twofold: 1) Prevent collisions between stations using the node, and 2) Allow additional time for stations sending more data than for those sending little data.

A DAMA node (called a DAMA MASTER) is actually a LAN node with at least one link to a backbone network. A single DAMA master can support up to 16 TNCs (thus 16 radios), perhaps providing two or three backbone link frequencies, a couple of user frequencies, some frequencies for BBS systems, and so on. Each LAN has a DAMA master node, and the adjacent DAMA masters are linked together. This inter-node link uses NET/ROM protocol, exactly like the networks just discussed.

Within a LAN, users connect to the DAMA master. Through special coding in the AX.25 frame, the user's TNC is placed into a "DAMA SLAVE" mode. From this point on, the user's TNC will not transmit any packets until the master has instructed it to do so by sending a poll frame to the slave. In this manner, no two slave TNCs will transmit at the same time, eliminating collisions.

The demand access feature of DAMA comes through intelligence built in to the master node. When the master polls a user, the user must respond - even if it's only an ack saying "I have no data to send". After polling station A, the master polls station B, then C, and so on. Any station that does not have data to

send has its priority reduced. The master then polls those stations with higher priority more frequently than those with low priority. As soon as a low priority station responds to the poll by sending data, the master bumps that user's priority back up to the maximum.

The DAMA system may provide an excellent alternative in areas where LANs are very large, resulting in many LAN users being unable to hear all other users. The major drawback to the DAMA system, at this point, is that the DAMA master station requires a computer at the node site to run the special software. In addition, multiple frequency operation requires one TNC for each frequency with special Dm firmware installed.

Users accessing DAMA systems should be using special DAMA slave TNCs. The DAMA software as distributed contains EPROM images for TNC-2 clones, enabling them to be DAMA slaves. These EPROMS use a slightly modified version of the WA8DED Host mode to communicate to the computer, and some software packages are readily available for use with these. Among the more popular programs in this category are Grafik Packet and ESKAY.

Kantronics will soon be releasing a new DAMA capable EPROM for some models of their TNCs. The new DAMA EPROM will allow users access to DA-IVIA networks as a slave using any terminal program, dumb terminal, or even Kantronics Host mode.

Conclusion

Networking in the amateur packet world can be considered to be still in its infancy. The first NET/ROM firmware was released around 1987, and many improvements have followed. Other networking systems have been developed (ROSE and TCP/IP) for amateur use, but to-date, none has been declared the clear "winner" for packet networking.

Debates rage about the "best" or "perfect" network, but if such a system already existed, world-wide VHF packet would be a reality instead of a dream.

K-Net and KA-Node are trademarks of Kantronics Co., Inc.

NET/ROM is a trademark of Software 2000, Inc.