the TEXNET packet-switching network

part 1: system definition and design

Four-node digipeater system reduces congestion, speeds packet delivery

In response to the phenomenal growth of packet radio over the past three years, many packet repeater ("digipeater") networks have been developed, allowing packet communications to be extended over many hundreds, even thousands, of miles. The operation of these digipeater systems has not been without some significant problems, however; most notably, congestion and difficulty in maintaining connections through more than about four or five individual repeaters, with excessive time delays between endpoints.

In an effort to resolve these problems, we decided to establish a rapid, reliable network that would allow Texas packet radio operators to communicate effectively over distances of several hundred miles *in real time.* We now have TEXNET, a four-node network with some of the communication trunks between nodes operating at 9600 bits per second.

In developing TEXNET, our goal was to minimize the cost of building a network node, yet provide very small transmission delay time between users. After the system was in place, we added additional services to the network without degrading the quick response time.

digipeaters — pro and con

A digipeater repeats what is transmitted to it; it can't remember anything about what it is repeating.

A good analogy to a "string" of individual digipeaters is a bucket-brigade line at a fire, in which *each person* is handed a bucket, which he or she in turn hands down the line to the next person. Eventually, each bucket makes it to the last person in the line, who throws the water onto the fire. With digipeaters, the system works like this: the first person in the line fills up a bucket and hands it to the second person. The second person hands it to the third, and so on until it reaches the end of the line (the receiver). Utilizing digipeaters, the sender must wait until the bucket is delivered to the receiver, emptied, and then sent backwards up the line back to the sender, who fills it up again. In other words, *there's only one bucket!*

Just as water can leak or spill from the bucket each time it's passed in the bucket brigade, data packets can be lost at each digipeater. Thus, it's not at all certain that all of the packets will arrive at their appointed destination.

On packet radio we use a layer 2 protocol called AX.25 to assure that all the packets get to the destination in the right order, without any getting lost along the way. This protocol is no more than a set of rules upon which the sender and receiver have agreed; one of the rules is that the receiver will "acknowledge" packets when they're received. The receiver sends these acknowledgments ("ACK," for short) backwards up the bucket-brigade line (i.e., the string of digipeaters) to the sender. If the sender doesn't see an acknowledgment within a few seconds, it assumes that the packet was lost somewhere and retransmits the packet. When the ACK is received, the sender transmits the next packet. However, only one bucket can be put into the line at a time; the ACK must come back from the receiver before the next packet can be started. Notice that none of the digipeaters really get involved in what's going on; they merely repeat the packets. This method of acknowledgment is known as *end-to-end acknowledgment* — that is, the acknowledgment travels all the way through the string of digipeaters from the packet receiver back to the packet sender. (AX.25 is really a little more complicated than this, but it's a good approximation of what's happening.)

Thomas H. Aschenbrenner, WB5PUC, and Thomas C. McDermott, N5EG, Texas Packet Radio Society, 265 Daniel Drive, Plano, Texas 75074

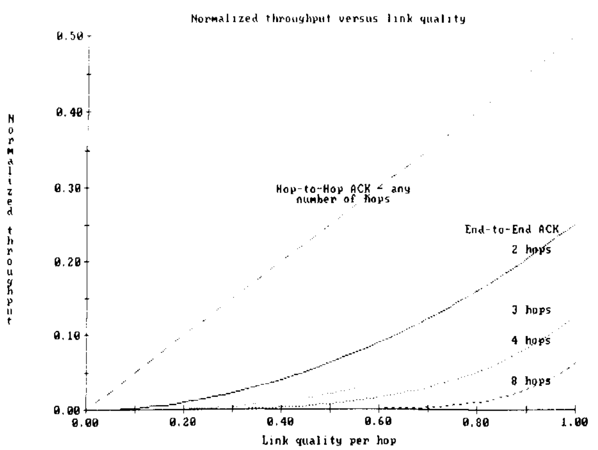


fig. 1. Throughput vs. rf link quality end-to-end and hop-to-hop methods. This graph shows the relative throughput for several packet repeating strategies. The throughput for hop-to-hop acknowledgments is independent of the number of hops, while the throughput for end-to-end acknowledgments degrades with an increasing number of hops. Both strategies degrade (i.e., the throughput is reduced) as the quality of the packet links becomes worse and causes more retries.

As anyone who's used a string of digipeaters to communicate with another station can attest, AX.25 works. But because we have only one bucket, the throughput (the amount of water that can be delivered to the fire) is very limited, and the greater the number of digipeaters in the path, the worse the problem becomes. In fact, it gets *much* worse *very* fast. Since the loss of a packet or of an acknowledgment at any point in the path will cause the retransmission of a packet, the probability of both the packet and the acknowledgment making the round trip successfully quickly becomes very small. This means that communicating a single packet will require many retransmissions, so throughput is reduced significantly.

A better method of relaying the information along a network would be to have each repeater along the way check the validity of the information before passing it on to the next repeater. That is, each repeater would ask for a "fill" of the message before sending it down the line. When the sender is assured that the first repeater received the packet, it could immediately send the next packet into the bucket-brigade line. Thus, we would have a bucket-brigade line with many buckets. Once the first bucket is delivered to the first repeater, another bucket would be filled and delivered to the first repeater by the sender. Thus the throughput (amount of water delivered) would be increased greatly. If we were to employ this strategy in relaying a message, the chance of losing packets grows only slightly larger as the number of digipeaters is increased. This method is called *hop-to-hop acknowledgment*, as each packet is acknowledged between adjacent repeaters before being sent along the network. As the probability of losing a packet grows, the necessity of retransmitting it increases — that is, fewer packets per unit of time can be transmitted. **Figure 1** compares the throughput for hop-to-hop and end-to-end ACK methods to the rf path quality between each repeater.

Response time — the amount of time it takes for a message to be delivered from the sender to the receiver, and for the sender to receive the ACK — is an additional consideration. **Figure 2** compares the round-trip response time for hop-to-hop and end-to-end ACK methods to the rf link quality between each repeater. As can be seen, if the repeaters operate virtually error-free, then the end-to-end acknowledgment strategy works very well. However, if the quality is degraded even slightly, it can be seen that the end-to-end strategy behaves poorly, whereas the hop-to-hop acknowledgment degrades linearly only with path quality. It should be noted that 2-meter packet users consider a path with 75 percent reliability extremely good!

A second problem with any string of digipeaters lies in determining just where a problem exists. If one of the repeaters in the string isn't receiving packets at all, then the sender and receiver know only that the path is "blocked" and are unable to tell where the packets aren't being relayed.

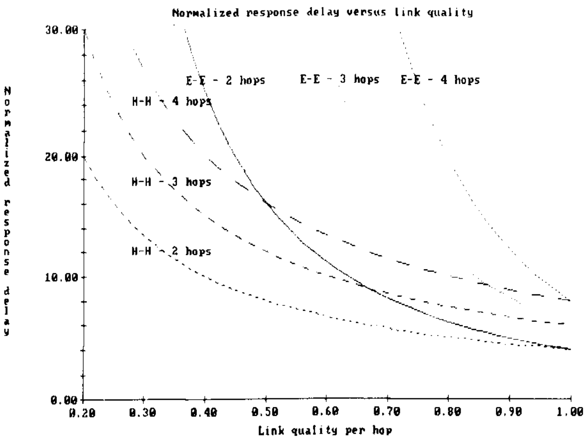


fig. 2. Response time vs. rf link quality end-to-end and hop-to-hop methods. This graph shows the relative response delay for several packet repeating strategies. The response delay for hop-to-hop acknowledgments is shorter (faster response) than for end-to-end acknowledgments unless the rf links are perfect (quality = 1.0), when the two have equal performance. Both strategies degrade (i.e., the response slows down) as the quality of the packet links degrades and causes more retries.

a network solution

To try and solve some of these problems, we wanted to build a packet network that would acknowledge packets at each step on the path, operate with minimal time delay, and provide us with information about the network: specifically, a measurement of the path quality at each point in the network and clear indication of where the break in the path has occurred, should one of the paths be out or one of the nodes be broken. It could also provide other features, such as conference bridges between any three or more users, or bulletin board service to several users simultaneously.

Earlier we stated that AX.25 would provide only end-to-end acknowledgments. This is because X.25 (from which AX.25 was derived) was designed basically as a point-to-point protocol. As a result, it works very well when Station A wants to communicate reliably with Station B. Our network, however, must use some additional strategies (protocols) for managing things like supervision (altering routing tables, reinitializing nodes), error recovery (to indicate where network has failed), and hop-to-hop acknowledgments.

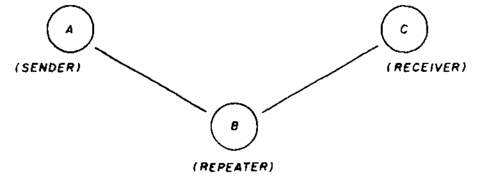


fig. 3. This is a map of a simple network. Station A is sending packets to station C through an intermediate repeater (station B) because stations A and C are too far apart to communicate directly.

It's at this point that we'll break up the problem of communicating between two stations into several "pieces," each of which will have the responsibility of solving only a part of the total problem. If we're smart about how to divide up the problem, each piece will be a fairly straightforward design problem, and each piece will know what to expect of the other pieces. That is, each of the pieces will cooperate with the others in order to solve the entire communication problem. This approach is called "layering" a problem.

layered protocols

Let's look at the problem of communicating a message along a network. Station A is the sender, Station B is the repeater, and Station C is the receiver (see **fig. 3**). The sender, Station A, needs a way to send information along the route A-B-C.

The first problem is to make sure that the information gets from A to B accurately. Let us assign this problem to layer 2 (ignoring layer 1 for now). That is, layer 2 must get information from A to B in the correct sequence, without duplicating any packets and without losing any packets. AX.25 works just fine for this job. Getting data from A to B is a point-to-point problem; A sends the packets and B acknowledges them. Now that some packets have traversed from A to B, how does B know what to do with them? This is a job for the next layer of the protocol, layer 3. Layer 3 tells each node where the information is going; if B is unable to send the information to C (or a path that leads to C), then it informs A that something is

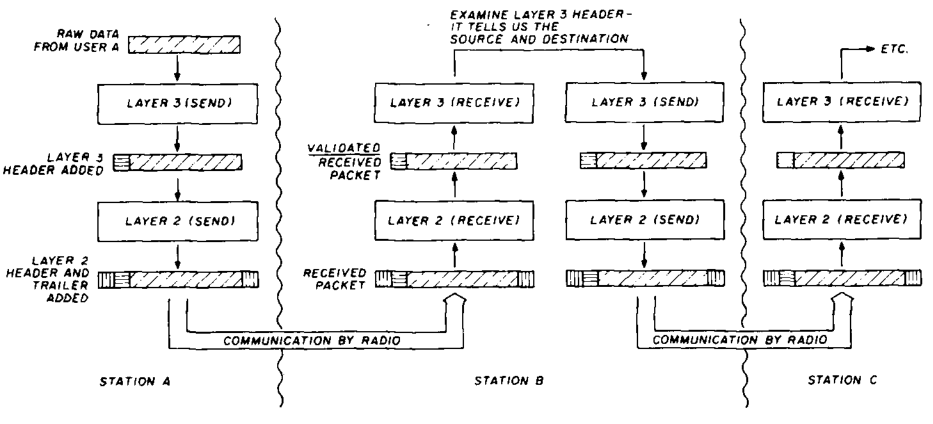


fig. 4. Layering the protocols (wrapping and unwrapping). Each node contains software to wrap the layer 3 and then the layer 2 information around the data. Each node also contains software to unwrap layer 2 and layer 3 information from the data, and examine this information to determine how to route the data to the final receiver.

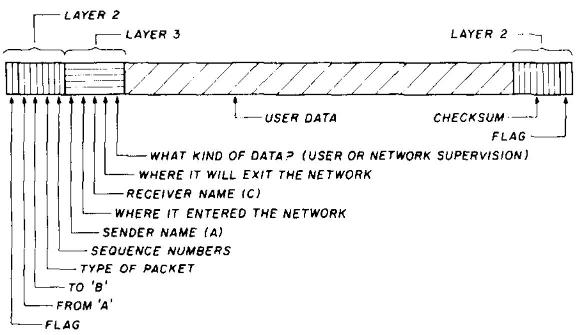


fig. 5. Contents of the packet. The layer 3 envelope is wrapped around the data first. It tells each node where the data came from, and where it is going, and the network entry and exit points. The layer 2 envelope is wrapped around the layer 3 envelope, and tells two adjacent nodes how to exchange the information reliably between themselves.

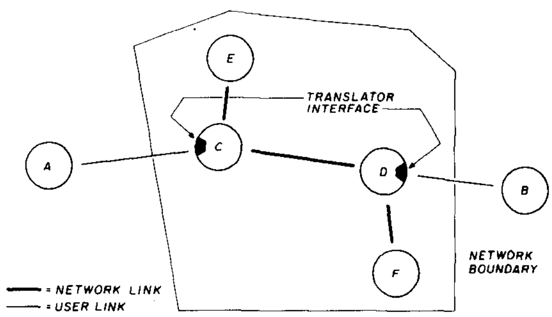


fig. 6. Drawing the network boundaries—which nodes translate from AX.25 to TEXNET-IP. In order for network users not to have to understand the internal network protocol, each network node has a user entry point, which supplies an English-language interface between the user and the network. The user may ask the network for services via this interface.

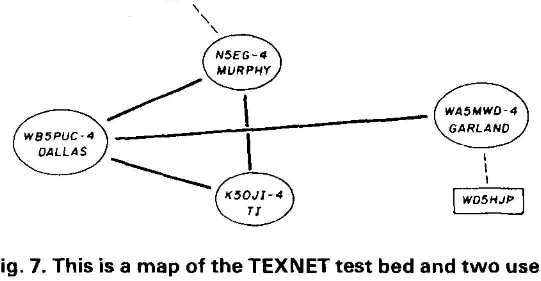


fig. 7. This is a map of the TEXNET test bed and two user stations (operators, TNCs, and 2-meter radios) of the network. Network trunks exist between Murphy, Dallas, and TI, but Garland can only communicate with Dallas.

wrong with the network. So Station A has to add a little additional information at the front of each packet that tells the intermediate stations where the information came from and where it's going.

Let's examine the sequence of events that occurs here. In **fig. 4**, Station A generates some data and sends it to its own layer 3 box. This box adds some information to the data packet (who the sender and receiver are, for example). Then the layer 3 box gives this slightly larger packet to the layer 2 box, which in turn adds a little information to it (things like a checksum for detecting errors, and the callsigns of Stations A and B). Layer 2 at Station A then assures that this packet is reliably delivered to the layer 2 box at Station B. The layer 2 box at Station B, happy with this packet, "unwraps" the layer 2 information and delivers what's left to the layer 3 box at Station B. The layer 3 box at Station B now looks at the information that the layer 3 box at Station A added to the packet and decides what to do with the packet. Probably Station B will determine the best way to get to Station C, and will tell its own layer 2 to send this packet to Station C; Station B will not alter the layer 3 information that station A put on the packet. Then the layer 2 box at Station B will add information (like a checksum, and the callsigns of Stations B and C) to the packet, and reliably deliver it to Station C. The "unwrapping" (examination of the layer 3 header, and the "rewrapping" of the layer 2 data) will continue at each node until the packet arrives at C. At Station C, the layer 2 box will "unwrap" the layer 2 data and then present the remainder to the layer 3 process, which will notice that this packet is destined for this station. Then the layer 3 box at Station C will remove the layer 3 information and present the raw data to the receiver at Station C. The contents of this individual packet is shown in **fig. 5**.

Thus raw data has traversed the network from A to B, through intervening users. At each step of the way it was error-checked and reliably exchanged by adjacent nodes, and each node decided how to route the information along to the final destination. Thus we have built a method that offers hop-to-hop acknowledgment, routing information reliably between two points. It also returns error messages to the sender, since it knows who the sender and receiver are.

There are two important points to consider: a standard protocol (AX.25) has been used at layer 2, and some of the more distressing problems with digipeaters have been solved. Unfortunately, we've added the requirement that the sender and receiver, Stations A and B, understand and implement an additional protocol, the layer 3 box. Rather than require this, we can instead build a "translation" function into Stations C, D, E, and F. These would converse with A and with B in an English language-like manner, and would make all the decisions about to and from whom packets should be delivered. Thus if Stations A and B can wrap and unwrap the layer 2 information from each packet, and if the human operators at Stations A and B understand the English language commands that C through F need in order to translate and add layer 3 information to each packet, then the users at A and B need only to possess a TNC that has a layer 2 function that is compatible with AX.25 (see **fig. 6**). Fortunately, all TNCs are capable of this.

the TEXNET implementation

This is how TEXNET operates. A user connects to TEXNET just as anyone with a TNC would connect to any station. For example, let's look at the sequence K5YEF (in Plano) would follow to utilize the network to talk to a station in Garland (see **fig. 7**).

In this case, K5YEF is located near the Murphy node, and WD5HJP is located near the Garland node. Notice that the network node stations are not normal TNCs, but are TEXNET network nodes instead.

What K5YEF types is shown underlined; all other text appears on his CRT.

```
CMD>C N5EG-4
CMD>***CONNECTED TO N5EG-4

N5EG-4 VIRTUAL CONNECTION 03 AT 17:04:57 ON 11/26/86

   ***  WELCOME TO TEXNET    V0706-WB5PUC  ***

COMMAND ? Circuit WD5HJP @ GARLAND
YOUR CONNECTION IS ESTABLISHED
```

From this point on, the communication proceeds normally.

What does the station WD5HJP see? Let's take a look at WD5HJP's CRT.

```
CMD>***CONNECTED TO WA5MWD-4
INCOMING TEXNET CONNECTION FROM K5YEF-0 AT MURPHY
```

At this point, whatever K5YEF has typed appears on the screen.

The users of TEXNET connect to it on 145.05 MHz, at 1200 Baud using their standard TNCs. The network communicates between its own nodes using AX.25 as the layer 2 protocol and TEXNET-IP as the layer 3 protocol. The network nodes run their inter-nodal trunks at 9600 Baud on either 220 or 450 MHz, or can run them at 1200 Baud on 2 meters.

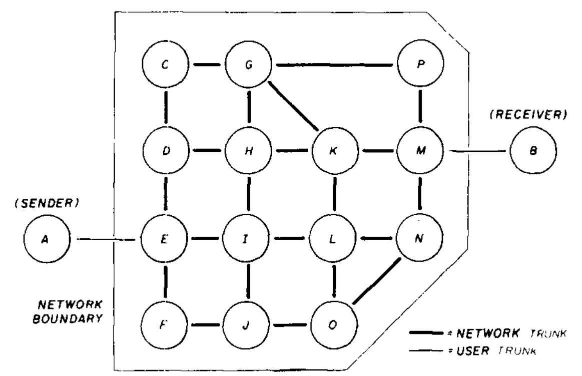


fig. 8. A Mesh network topology. Many paths through this network are possible, AEILNMB, AEDHKMB, AEIHKMB are 3 possible routes. This network is resistant to failures, but expensive to implement.

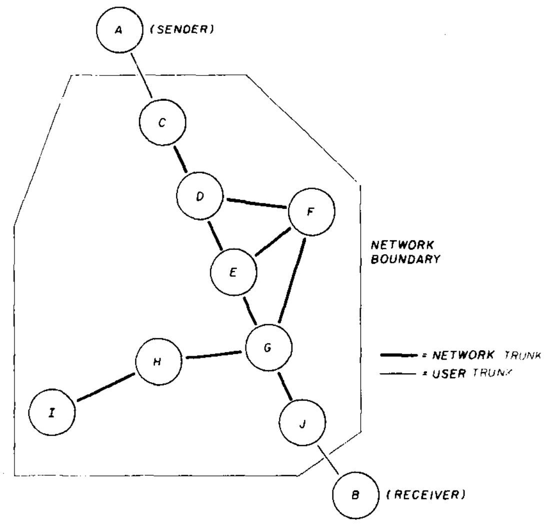


fig. 9. A Backbone network topology. A few alternate routes exist through the network, but single-point failures are possible. This is a low-cost network.

It would be best if the users of this network (Stations A and B, for example) had a way to communicate with the network that didn't require the use of human operators and English language commands. Then computers (at A and B) could control setting-up and tearing-down connections through the network. Unfortunately, this type of layer 3 protocol — outside the network — requires that all TNCs be standardized for a layer 3 communication process, and no standards now exist in the Amateur community for this function.

The TEXNET-IP layer 3 protocol is "hidden" from all the users because the entry and exit nodes of the network translate the instructions from the users from English to TEXNET-IP and back again. The TEXNET-IP is utilized only within the network, and it is of a family of network protocols known as "datagram" (that is, each packet carries all of the information needed by the network). The TEXNET-IP protocol adds 5 bytes of overhead to the front of every packet inside the network, but is not suited for use as a user layer 3 protocol.

network topologies

How should all of these network modes be physically located? How should the communication paths between nodes be set up? The topology of a network is a map of the network — that is, where the nodes of the network are located, and which nodes are within rf range of other nodes. The topology defines which nodes can be connected to each other, and gives a name to the different types of network configurations that could be made.

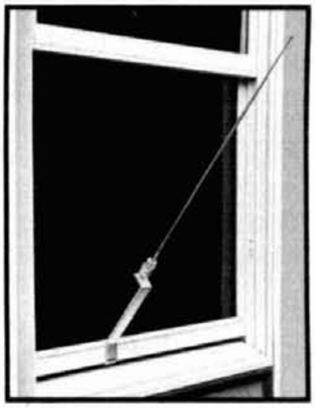
There are many topologies available for setting up a network, but we'll look at two common ones here. One way to set up a network — a "mesh" network — is shown in **fig. 8**. Mesh networks have many nodes, and many possible ways to route information between two users. Meshes also have a lot of "resilliency." They can suffer outages of nodes and/or paths, yet still have a way to route information between any two points.

Because the Texas Packet Radio Society doesn't have enough money to build and install switching nodes everywhere, we've chosen a topology that minimizes cost, but unfortunately degrades the survivability of the network. In our network, we've installed a "backbone" arrangement as shown in **fig. 9**. In this topology, nodes are installed along a "skinny" route between the major population centers — those users with the largest amount of traffic to send a long distance. Alternate routes to some of the paths are included. Each of the nodes contains a "table" in memory which is a map of the system, so that it knows to which node packets should be forwarded, depending upon which node will receive the packet and deliver it to the final user. These tables contain alternate routes in case the primary route is unavailable. In addition, each node contains an area in the memory where the routing table can be "patched" to accommodate recent changes to the map. These recent changes can be loaded into the network nodes by the network control operator. This type of routing is known as *static* or *directory* routing.

Further articles in this series will focus on specific issues addressed in implementing the TEXNET network. One section will be devoted to the hardware that was designed, and one section will be devoted to the software that was designed (protocol layers). The software section will also describe additional features provided by the network.

ham radio

PORTABLE ANTENNA



MODEL AP-10

Designed for
APARTMENTS
MOTELS
VACATIONS

PRICE
$51.95

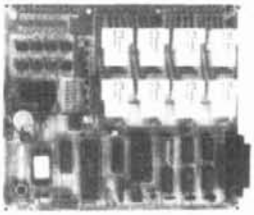
Add $3.00
Shipping and Handling

Quick Simple Installation. Operates on 2, 6, 10, 15, 20, 30 and 40 meters. All coils supplied. Only 22-1/2 inches long. Weighs less than 2 lbs. Supplied with 10 ft. RG 58 coax and counter poise. Whip extends to 57 inches. Handles up to 300 watts.
VSWR — 1.1:1 when tuned
Write for more details and other B&W products

ALL OUR PRODUCTS MADE IN USA
B·W **BARKER & WILLIAMSON**
Quality Communication Products Since 1932
At your Distributors. Write or Call.
10 Canal Street, Bristol PA 19007
(215) 788-5581

I/O FOR REAL WORLD CONTROL

$199.95 CALL OR WRITE FOR MORE INFORMATION



MODEL CIP/35A SHOWN WITHOUT COVER

NOW ANY PERSONAL COMPUTER CAN HAVE THE MOST COST EFFECTIVE AND VERSATILE I/O BOARD ON THE MARKET TODAY!

- Serial Link Interface RS-232 or TTL
- 8 Relay Outputs, High Current AC/DC Form A & C
- 8 Opto-Isolated Inputs Plus 8 Bit Counter
- 8 Bit A/D with Span Adjust 0 to 5V, Provisions for up to 8 Input Channels
- EASILY Programmed & Controlled Using BASIC Statements
- Perfect for Lab Work, Machine Control, Security Systems, & Data Acquisition
- Unprecedented Usability as Attested by University, Government & Industrial Users
- Complete Documentation with Software Examples & Total Engineering Support

• IBM • HP • RADIO SHACK • COMMODORE*

* ALL REGISTERED TRADEMARKS

SIAS Engineering, Inc.
831 S. POWERS RD. / SALINA, KS 67401 / (913) 823-9209

the TEXNET packet-switching network

part 2: hardware design

In last month's article,[1] we discussed network algorithms and the software layering. This month, we'll focus on the design and testing of hardware for the network, and on the results that have been achieved to date.

system partitioning

Partitioning hardware to minimize the number of signals that must connect between units offers three benefits: simplified cabling, easy measuring and modification of individual units, and flexibility in the construction of the network.

Figure 1 is a block diagram of a TEXNET network node. There are four main pieces: a local area network (LAN) radio, which in this case is a 1200-baud AFSK modem and 2-meter radio; an inter-processor (IP) radio (a 9600-baud FSK radio and modem) for use as the high-speed network trunk; a node control processor (NCP) that contains the microprocessor and communications ICs; and the power supply, which contains a three-state float charger, battery, and circuitry for automatic uninterrupted power should ac power fail.

LAN radio and modem

This channel is the primary method by which users with TNCs and 2-meter radios connect to a network node. By connecting the modem separately from the processor, any modem can be used — 300 baud, 1200 baud, 2400 baud, or whatever might be desired. **Figure 2** is a diagram of the modem, which is similar to the TAPR TNC-1 modem. We chose to implement an active filter equalizer with op-amps instead of a switched-capacitor filter IC. The modem includes a 45-second time-out timer to disable the transmitter should the controller fail for some reason. The strap allows setting the EXAR demodulator VCO center frequency, but better results can be obtained by adjusting the VCO frequency control pot and observing the received "eye" pattern on an oscilloscope from an AFSK signal known to be good. An eye pattern is observed on an oscilloscope by synchronizing the scope trigger with the recovered clock and displaying the data. Since the data displayed is not involved in triggering the scope, a random display of all data sequences is shown, but the zero crossings are fixed in time on the screen, yielding an open area in the center of a data bit (known as the "eye"). **Figure 3** shows a typical eye pattern, the recovered clock, and the slicing level (which decides between a 1 and a 0). The basic decision circuit is shown in **fig. 4**.

The radio is an ICOM-IC22S, a popular 2-meter transceiver, with the frequency hard wired to 145.05

Thomas H. Aschenbrenner, WB5PUC, and Thomas C. McDermott, N5EG, Texas Packet Radio Society, P.O. Box 831566, Richardson, Texas 75083-1566

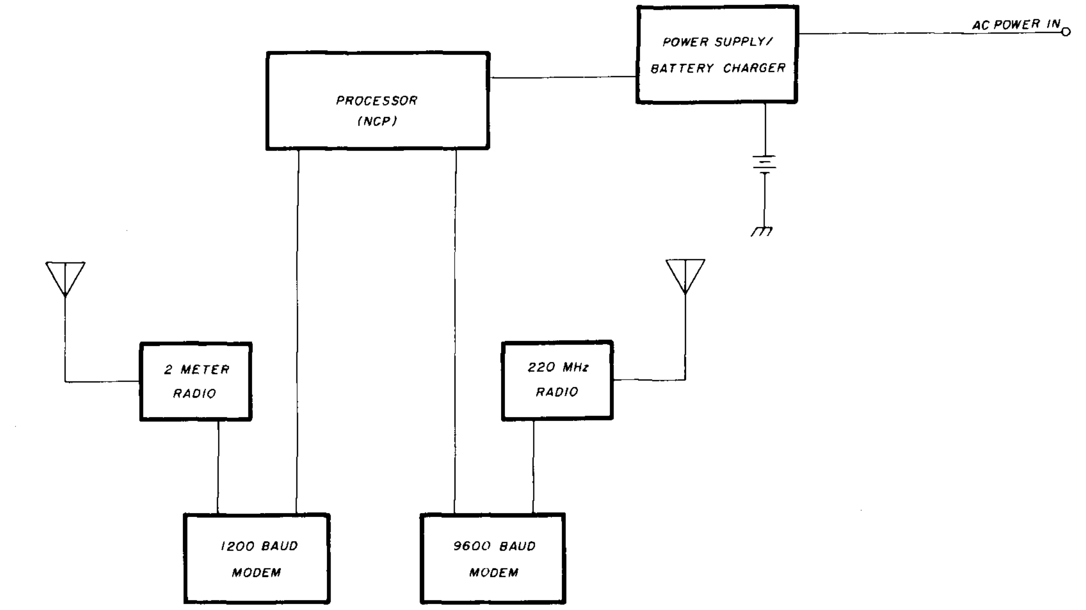


fig. 1. Hardware block diagram of TEXNET node. This diagram shows the partioning of the node into functional blocks, the processor, the two modems / radio interfaces, and the power supply / battery charger.

MHz. The transmit audio is injected after the microphone amplifier; the receive audio is tapped off prior to the audio PA in order to avoid the severe degradation of frequency response that results if the speaker and microphone leads are used for audio pick-off and injection.

IP radio and modem

The performance of the network trunks is very important in determining the overall throughput of the network as a whole. As each user sends traffic into a network node, all traffic is multiplexed (combined) onto the high-speed trunks. Thus the trunks carry a much greater amount of traffic than the user links. Because of this, we have decided to operate the trunks at 9600 baud, with rapid Transmit/Receive (T/R) switching. Rapid T/R switching is required because at 9600 baud, actual packets take relatively little time to transmit, and the T/R delay can determine the effective channel capacity.

Figure 5 illustrates the effective channel capacity versus T/R delay for a 9600-baud channel with no errors, and one acknowledgment packet for the entire transmission. Several different values are shown: one indicates the number of packets per transmission, another the number of bytes per packet, and a third for two values of DWAIT (digipeater waiting time, which allows a digipeater transmission priority). Because there are no digipeaters in TEXNET, DWAIT = 0. (A value of DWAIT = 80 ms is typical for 2-meter channels.) Our experiments involved the use of a pair of Hamtronics FM-5 220-MHz fm transceivers and K9NG's modems. These radios are modified to operate FSK, and the received data signal is tapped off the quadrature detector in the receiver. Since these radios are PIN-diode switched between transmit and receive, we were able to make them operate with 40 ms T/R delay, although in practice 80 ms was allowed.

We encountered some difficulty in making the radios operate properly at 9600 baud. Apparently these problems were due in large part to variations in the performance of different FM-5 radios, which were designed not for high-speed data operation, but rather for fm voice operation. In an effort to improve the operation of the radios, a number of experiments were run, and modifications were made to the modem.

To understand this better, let's review the basics of frequency-shift-keyed (FSK) data transmission, the spectrum of a non-return-to-zero (NRZ) data signal at baseband, and the performance of i-f filters in the time domain. We'll see that all three have to be addressed properly to assure proper performance of the radios and modems.

An NRZ signal is one that toggles between logic 1 and logic 0 no more than once per bit period (see **fig. 6**). In an FSK system, two frequencies are transmitted — one for logic 1, the other for logic 0. At the output of the discriminator/quadrature detector at the receiver, the two frequencies are translated back to voltages. If the frequency of the transmitter were to

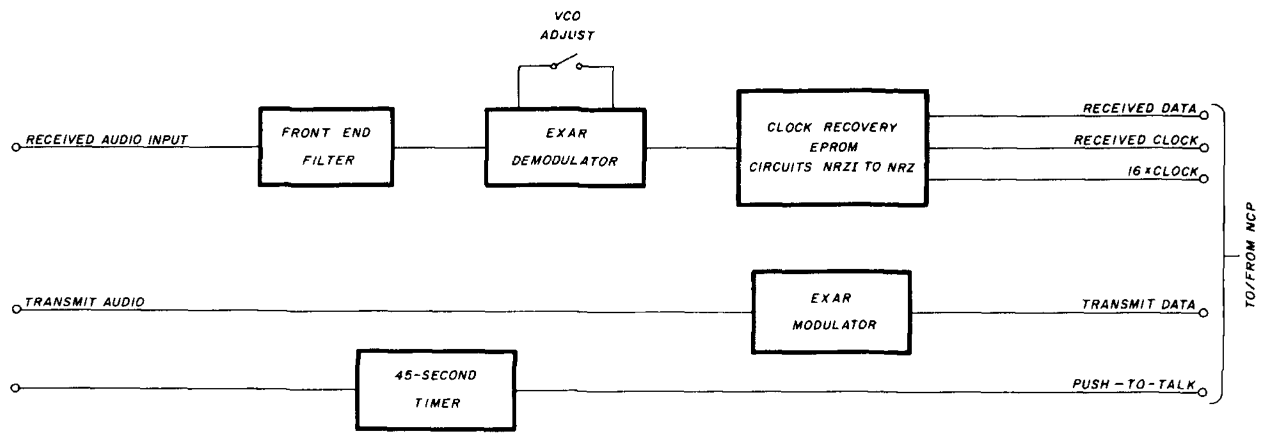
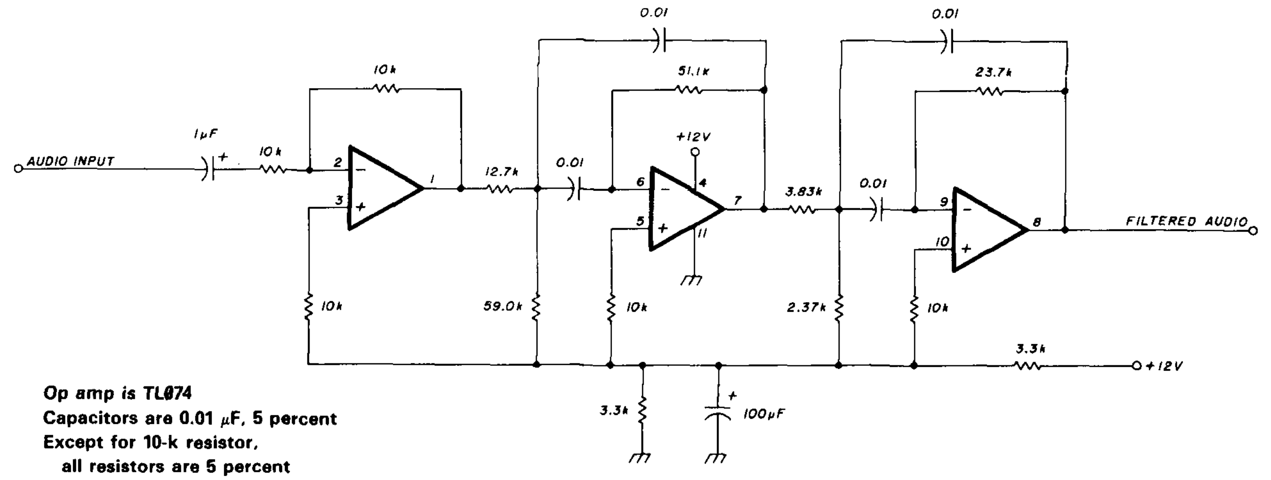


fig. 2A. 1200-baud AFSK modem.



Op amp is TL074
Capacitors are 0.01 μF, 5 percent
Except for 10-k resistor,
all resistors are 5 percent

fig. 2B. Front-end active filter schematic.

vary slightly, then both the logic 0 and logic 1 voltages would also vary. A simple method to determine the correct "slicing level" for deciding between a logic 1 and a logic 0 is to choose the voltage halfway between the two. This is simple to do by using a low-pass filter with a time constant quite a bit longer than the data period to generate the slicing voltage level. Then the slicing threshold will "track" the logic 1 and 0 voltages automatically (see **fig. 7**). This requires, however, that the transmitted data have, on the average, the same number of ones as zeros; if they don't, the recovered slicing level will be biased off the true center point (**fig. 7**).

In HDLC, the code used for AX.25, there is no guarantee that the code will be dc-balanced (i.e., have same number of ones as zeros). In fact, the flag character contains two zeros and six ones, thus having a large dc offset from one-half. A simple way to solve the problem is to use a pseudo-random scrambler to cause some apparent randomization of the data sequences. A self-synchronized descrambler is used on the receiving modem to recover the original bit stream. This is the method used on the K9NG modem to send and receive data, a 17-stage scrambler being used. With this arrangement, the average number of ones and zeros is nearly the same. Certain sequences into a scrambler can, however, produce long strings of ones or zeros. If a long string were to occur, our low-pass filter in the receiver would tend to drift off the center voltage, halfway between the 0 and 1. So we must compromise the time constant of the low-pass filter in the receiver slicing level circuit (which we would like to make very long) with the need for rapid T/R switching, where we need to acquire the proper level quickly. In addition, any capacitors used to couple the analog signal must have long time constants; if

SPECIAL

SYSTEM #1
$399.00
Motherboard with bios and first 64K of RAM. Upgradable to a full 640K of RAM. Flip top case. K8XT (at look alike) keyboard. 150 watt power supply with all the power needed to run extra drives and cards.

SYSTEM #2
$699.00
Motherboard with bios and first 256K of RAM. Upgradable to a full 640K of RAM. Flip top case. K8XT (at look alike) keyboard. 150 watt power supply. Dual Disk Drive card with cables. One floppy drive. DS DD 360K. A color graphics card with RGB and composite output.
(All you need is a monitor)

SYSTEM #3
$999.00
Motherboard with bios and first 256K of RAM. Upgradable to a full 640K of RAM. Flip top case. K8XT (at look alike) keyboard. 150 watt power supply. Color graphics card with RGB and composite outputs. Multi I/O card with two disk drive ports, one parallel port, one serial port and one serial port option, one game port, clock and calendar with battery backup. Two floppy disk drives DS DD 360K and a composite monitor.

SHIPPING INFORMATION: PLEASE INCLUDE 10% OF ORDER FOR SHIPPING AND HANDLING CHARGES (MINIMUM $2.50, MAXIMUM $10). CANADIAN ORDERS, ADD $7.50 IN US FUNDS. MICHIGAN RESIDENTS ADD 4% SALES TAX. FOR *FREE* FLYER, SEND 22¢ STAMP OR SASE.

✓ 183

HAL-TRONIX, INC. (313) 281-7773
12671 Dix-Toledo Hwy
P.O. Box 1101
Southgate, MI 48195
Hours
12:00 - 6:00 EST Mon-Sat
"HAL" HAROLD C. NOWLAND
W8ZXH

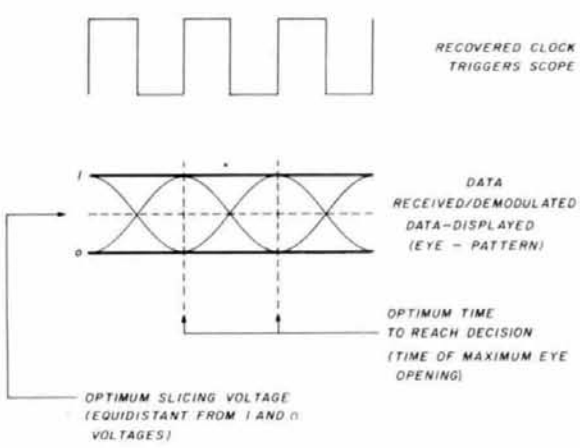


fig. 3. "Eye" pattern, recovered clock, slicing level. This figure shows what an "eye" pattern looks like, and how the oscilloscope is set up to make the measurement. The one/zero decision is made at the slicing voltage level at the instant in time of the rising edge of the recovered clock.

WOW! AT LAST!! A VERY AFFORDABLE COMPUTER AT A VERY AFFORDABLE PRICE

POWERFUL FULLY PROGRAMMABLE WITH 2K OF MEMORY—PORTABLE—6-7/8 × 1-3/8 INCH MODULE SINGLE-KEY ENTRY COMMANDS—DURABLE 40 KEY MEMBRANE TYPE KEYBOARD—Z80A BASED FOUR CHIP DESIGN—EDUCATIONAL—UNIQUE SYNTAX-CHECK REPORT CODES FOR ERROR IDENTITY—GRAPH DRAWING AND ANIMATED DISPLAY—ACCURATE TO 9 1/2 DECIMAL PLACES FOR FULL RANGE MATH AND SCIENTIFIC FUNCTIONS—AT AN AFFORDABLE PRICE

WE CANNOT TELL YOU THE MAKE OF THE COMPUTER BUT IT WAS MADE BY A FAMOUS WATCH COMPANY. THEY USED TO SELL FOR $99.95

WE BOUGHT OUT WHAT THE FACTORY HAD LEFT IN STOCK AND HAD TO REMOVE THE LABELS. THESE UNITS ARE UNPACKAGED LESS THE 9V WALL ADAPTER AND MANUAL. BECAUSE THIS IS A DISCONTINUED ITEM THERE IS NO WARRANTY

GET THEM WHILE THEY LAST — LIMITED SUPPLY

BUY 1st UNIT FOR **$19.95** BUY 2nd FOR **$16.95** 9V DC WALL ADAPTOR **$4.95**
BUY 3rd UNIT Non Operating For Parts **$10.95** MANUAL Over 150 Pages **$2.95**

See September 1984 issue of *73* for TIMEX/RTTY article

CHIP BONANZA (AT THESE PRICES THEY ARE A STEAL)

2708		$1.00 EA OR 10 FOR $ 9.00
2716		$2.25 EA OR 10 FOR $20.00
2732		$3.25 EA OR 10 FOR $30.00
2764		$4.00 EA OR 10 FOR $36.00
27128		$7.00 EA OR 10 FOR $60.00
6502		$4.95 EA OR 10 FOR $45.00
6810	(REG $3.95)	$1.95 EA OR 10 FOR $18.00
68A09	(REG. $19.95)	$5.95 EA OR 10 FOR $50.00
68A21	(REG. $9.95)	$2.95 EA OR 10 FOR $25.00
4116	(or equivalent)	8 FOR $ 5.99
4134		9 FOR $12.95
XR-2211	(SPECIAL)	$ 2.95
TMS 9900NL MICRO-P 64 PIN 8 BIT D/B + 16 BIT CPU		$ 4.95
TMS 9901NL MICRO-P PSI		$ 2.95
TIM 9904ANL MICRO-P CLOCK GEN. AND DRIVER		$ 5.95
TMS 9918ANL MICRO-P COLOR GRAPHICS AND DISPLAY		$ 9.95
KEYBOARD (9904) 48 KEYS MEASURE 4 × 10 (HI-TEK)		$ 9.95

APPLE II and APPLE II+ COMPUTER MAINFRAMES (fully populated) $150
Power supply, case and keyboard, separately available — Call or Write
Unit as described above, fully assembled & tested — $350 plus shipping
APPLE POWER SUPPLIES $29.95

Cassette Software:
I have cassette software ... send for list (reg. $9.95-$14.95)
Hal's Special Price: 1 for $3.95; 3 for $10.95; 6 for $19.50
10 for $29.50; 20 for $50.00
... or let Hal select 25 different cassettes (my choice) at 25 for $50.00!!!

SPECIALS

IBM COMPATIBLE COMPUTERS
Fliptop Case, keyboard, P/S and motherboard up and running. Call for details.
$399.00 ea + 5.00 S&H

LNA's
90° Kelvin, self-contained electronic polarator with feedhorn.
$59.95 ea + 5.00 S.H.
2/$100.00

SHIPPING INFORMATION: PLEASE INCLUDE 10% OF ORDER FOR SHIPPING AND HANDLING CHARGES (MINIMUM $2.50, MAXIMUM $10). CANADIAN ORDERS, ADD $7.50 IN US FUNDS. MICHIGAN RESIDENTS ADD 4% SALES TAX. FOR *FREE* FLYER, SEND 22¢ STAMP OR SASE.

HAL-TRONIX, INC. (313) 281-7773
12671 Dix-Toledo Hwy
P.O. Box 1101
Southgate, MI 48195
Hours
12:00 - 6:00 EST Mon-Sat
"HAL" HAROLD C. NOWLAND
W8ZXH

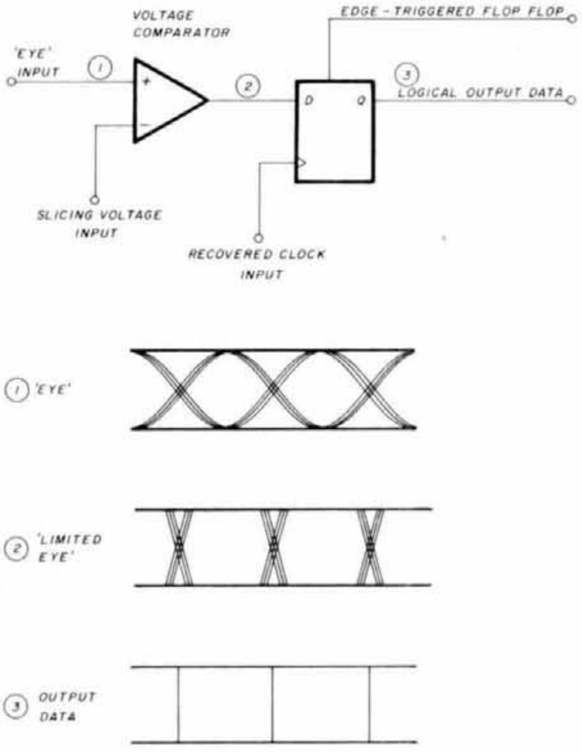


fig. 4. Decision circuit and levels. This diagram shows the decision circuit, basically a comparator and a flip-flop. The timing of the flip-flop clock and the reference voltage into the comparator determine the exact point the decision is reached.

they don't, the average voltage will drift as the low-frequency components from the scrambler charge and discharge the coupling capacitors.

Several modifications of the K9NG modem are associated with increasing the time constant of these circuits, where they're not critical to the T/R switching delay. A list of the modifications is included in **fig. 8.** The most significant improvement came with addition of a group-delay equalizer in the receive section of the modem. In order to appreciate the requirements for this, let's look at how filters work both in the frequency domain and the time domain. A typical i-f filter used in voice or CW work has very sharp "skirts." That is, the amplitude response decreases sharply from the filter center frequency. In addition, the response is relatively flat within the passband. This filter has a great deal of time delay distortion (see **fig. 9**). The time delay is minimum at the filter center frequency and rises sharply at both the upper and lower filter cutoff points. This causes no particular problems with voice, where the distortion of the waveform isn't important. But with wideband data signals, where we need to distinguish the value of data bits that are adjacent in time, the spectral energy nearer to the filter cutoff points will undergo a greater delay through the filter than the energy near to the center frequency. When this signal is converted to baseband (i.e., fm demodulated), it can be shown that frequencies near the fil-

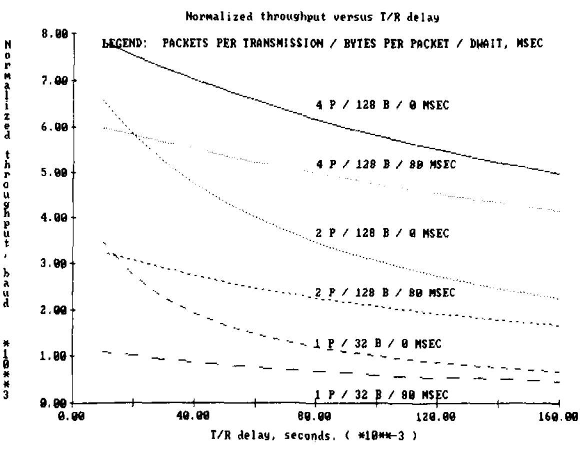


fig. 5. Effective channel capacity vs. T/R delay. This graph shows the effective capacity (baud rate) of a channel for different transmit/receive switching delays, with and without an additional delay, called DWAIT (digipeater wait time). TEXNET does not use digipeaters, so DWAIT = 0.

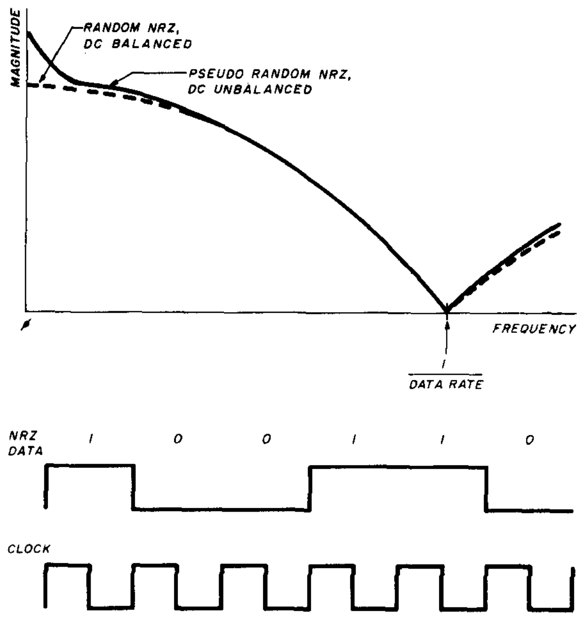


fig. 6. NRZ signal diagram. An NRZ signal has a value of logic zero or logic one for the entire period of the bit. The spectrum of a truly random signal is from dc, diminishing to zero at (1/bit time). A pseudo-random signal has a diminished spectrum near dc, theoretically vanishing.

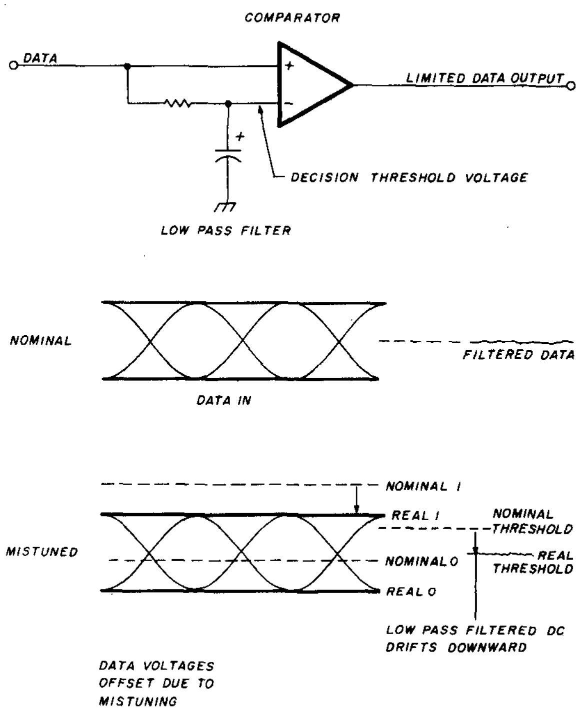
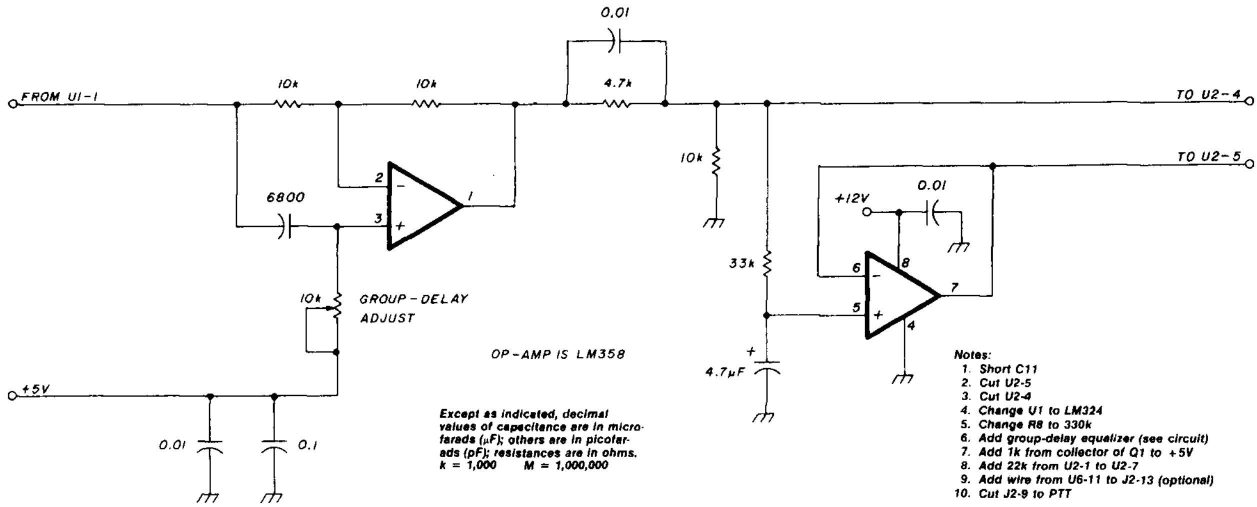


fig. 7. NRZ filtered, slicing level, low-pass filter to generate slicing level vs. frequency drift. If the NRZ is scrambled, so that the code is dc-balanced, then low-pass filtering the data will generate the correct logic-slicing voltage. As the one/zero voltages drift with receiver or transmitter mistuning, the slicing voltage will track automatically.



Except as indicated, decimal values of capacitance are in microfarads (µF); others are in picofarads (pF); resistances are in ohms. k = 1,000 M = 1,000,000

Notes:
1. Short C11
2. Cut U2-5
3. Cut U2-4
4. Change U1 to LM324
5. Change R8 to 330k
6. Add group-delay equalizer (see circuit)
7. Add 1k from collector of Q1 to +5V
8. Add 22k from U2-1 to U2-7
9. Add wire from U6-11 to J2-13 (optional)
10. Cut J2-9 to PTT

fig. 8. List of modifications to K9NG modem. The reference designators follow the TAPR documentation for the K9NG modem. The list includes all modifications to the modem, and additions (delay equalizer) to the modem circuitry. The schematic shows the additions to the modem.

ter center frequency correspond to the low-frequency spectrum of the NRZ (baseband) signal, while the frequencies near the filter cutoff (away from the filter center) correspond to the higher frequency components of the baseband signal. Thus we can "map" the time delay of the filter into baseband by "folding" the time response about the center frequency, which becomes the zero frequency of the baseband. Thus the filter produces small delay at low baseband frequencies, and larger delay at higher baseband frequencies.

The 455-kHz i-f filter in the FM-5 radio is a sealed ceramic unit; no adjustment is possible. Instead of designing a new filter with desirable time-delay characteristics, we instead built an active filter circuit, with flat amplitude response, but with an adjustable time-delay response. This network has maximum delay at dc, and decreasing delay at higher frequencies. The circuit is adjustable, so that we could construct an approximate inverse time delay to that caused by the i-f filter. **Figure 10** shows the amplitude, phase, and time response of this baseband group-delay equalizer (active filter), and the baseband eye pattern with and without the delay being equalized. After delay equalization, it can be seen that the eye is much more open near the center of the bit, when the 1/0 decision is reached. Our measurements indicated a 7-dB improvement. We also provided about 1 dB of peaking of the frequency response, which opened the eye about 1 dB more, yielding an 8 dB improvement in the receiver. Actual tests with the radios indicated that this made the difference between usable and nonusable performance. Without the equalizer, the radios

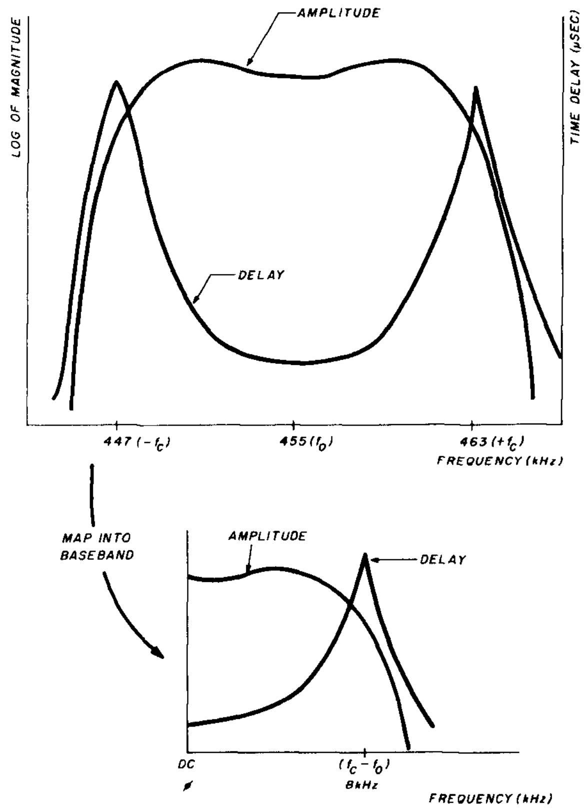


fig. 9. Group delay of 455-kHz i-f filter and map of delay into baseband by folding about F_0. (A) shows the group delay and amplitude vs. frequency for the 455 kHz filter; (B) shows how the band-pass response is mapped into the baseband by "folding" about the center frequency, F_0, of the filter (assuming symmetry in the filter).

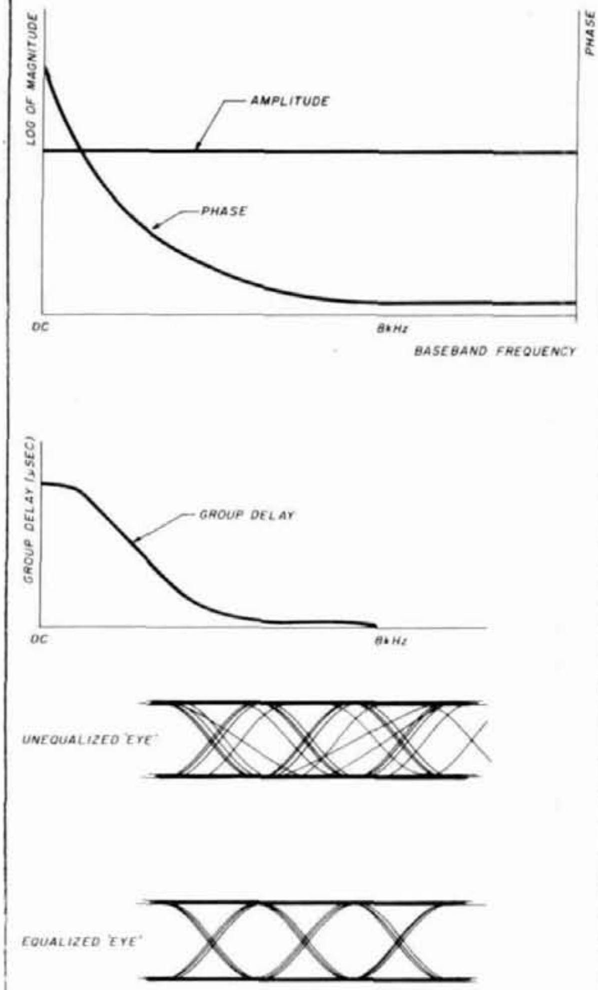


fig. 10. Amplitude, phase, and delay of the group-delay equalizer "eye" with and without equalization. The baseband group-delay equalizer has flat amplitude response, and a varying phase vs. frequency response. The group-delay is the negative of phase slope vs. frequency. Thus the group-delay is large as dc, and diminishes with increasing frequency.

"dribbled" (i.e., had a background error rate regardless of the strength of the received signal), which caused many packets to be lost.

One additional problem that had to be overcome on our "real" path test hop was insufficient image rejection in the FM-5. The test path is in an area where television channels 11 and 13 are very strong. The channel 11 video carrier is near the image frequency of our desired channels (the radio i-f is 10.7 MHz, so the image is 21.4 MHz lower than the signal frequency). **Figure 11** shows a simple filter; **fig. 12** shows its response (S11 and S12). This filter was extremely effective in eliminating the image response.

With these improvements, we've run a 12-mile path on 220.55 MHz at 9600 baud, at better than 98 percent

↙ 180

CONTINUOUS COVERAGE ANTENNAS FOR COMMERCIAL & AMATEUR SERVICE

Model AC 1.8-30 — 1.8 to 30 MHz

- SWR Max 2:1, 1.4:1 average from 1.8 to 30 MHz
- Can be installed in approximately 80 ft. space
- Ideal for commercial services for multi frequency operation without the need for antenna tuners or additional antennas
- Handles 1 KW, 2 KW PEP ICAS
- Higher power models available on special order. Contact your dealer or factory

$159.50 SHIPPING & HANDLING ADD $4.00

RF TRANSFORMER — DIRECT COAX FEED — 50 FT — 25-30 FT — 50 FT — BALANCING NETWORK

U.S. Patent No. 4,511,898

Model AC 3.5-30 — 3.5 to 30 MHz

- SWR less than 2:1 from 3.5 to 30 MHz
- Complete assembled. Balun terminated with standard SO-239 connector
- Power capability 1 KW - 2 KW PEP ICAS. Higher power model is available on special order.
- Designed for 50 ohm feedline
- Weather proof balun and balancing network

$167.50 SHIPPING & HANDLING ADD $4.00

Only 90 feet long

U.S. Patent No. 4,423,423

ALL OUR PRODUCTS MADE IN USA

B·W BARKER & WILLIAMSON
Quality Communication Products Since 1932
At your Distributors. Write or Call.
10 Canal Street, Bristol, PA 19007
(215) 788-5581

ALPHA DELTA
Model DELTA-4
Lightning Surge Protected
4 Position Coax Switch

Superior RF switching and equipment protection for Amateur, commercial and military communications stations. • Exclusive "center-off" ground position internally disconnects and grounds all antenna circuits for maximum protection when operator is away from the station. • Incorporates the famous Alpha Delta replaceable ARC-PLUG cartridge for active antenna protection while grounding other 3 antenna positions. An Alpha Delta FIRST! • Features custom designed cast aluminum housing with constant impedance micro-strip cavity construction for low loss performance from 1.5 to 450 MHz. No lossy wafer switch. • Positive detent ball bearing switch drive snaps firmly into place. • Handles full legal power • Built with pride in the U.S.A.

Model DELTA-4 4 POSITION COAX SWITCH (UHF CONNECTORS) $69.95

Available from your Alpha Delta dealer or order direct. Please add $4.00 shipping and handling. US only. Exports quoted.

ALPHA DELTA COMMUNICATIONS, INC.
PO Box 571
Centerville, OH 45459 (513) 435-4772
current solutions to current problems

I/O FOR REAL WORLD CONTROL

$199.95 CALL OR WRITE FOR MORE INFORMATION

MODEL CIP/35A SHOWN WITHOUT COVER

NOW ANY PERSONAL COMPUTER CAN HAVE THE MOST COST EFFECTIVE AND VERSATILE I/O BOARD ON THE MARKET TODAY!

- Serial Link Interface RS-232 or TTL
- 8 Relay Outputs, High Current AC/DC Form A & C
- 8 Opto-Isolated Inputs Plus 8 Bit Counter
- 8 Bit A/D with Span Adjust 0 to 5V, Provisions for up to 8 Input Channels
- EASILY Programmed & Controlled Using BASIC Statements
- Perfect for Lab Work, Machine Control, Security Systems, & Data Acquisition
- Unprecedented Usability as Attested by University, Government & Industrial Users
- Complete Documentation with Software Examples & Total Engineering Support

•IBM •HP •RADIO SHACK •COMMODORE*

* ALL REGISTERED TRADEMARKS

SIAS Engineering, Inc.
831 S. POWERS RD. / SALINA, KS 67401 / (913) 823-9209

✓ 181

INTN'L MONTHLY MAGAZINE-BY AND FOR ACTIVE RADIOAMATEURS

RADIOSPORTING

A magazine dedicated to quality and sportsmanship in amateur radio operating. Fresh, timely, practical and down to earth reading for little pistols and big guns. Written by world's best in their fields.

Includes DX News, QSL Info, 160 m, DXpeditioning, Propagation, Awards, contest rules and results, Traffic-Emergency, FCC News, New Products, Antennas, technical articles, equipment reviews and modifications, computer programs, Radio Funnies, Club Life, YL, RTTY, VHF/UHF, Mail Room, Classified Ads and much more in a magazine format with the speed of a bulletin.

RADIOSPORTING sponsors DX Century Award, Contest Hall of Fame, World Contest Championship and World Radio Championship contest.

"Your publication is superb! Keep it up!"
Joe Reisert, W1JR

"Your W2PV articles are priceless. Your magazine is super!"
Rush Drake, W7RM

"Let me congratulate you on a very impressive magazine. Just what I've been looking for as a DXer and Contester!"
Dick Moen, N7RO

"RADIOSPORTING, once received, cannot be tossed aside until it is read from cover to cover. Then reviewed again and again."
Chas Browning, W4PKA

Subscription rates: 1 year USA $18, Canada CDN$24, Overseas US$21; 2 years $33, $44, $39 respectively. Single issue $2.

TRY US! SUBSCRIBE OR SEND $1 FOR YOUR SAMPLE COPY.

✓ 182

RADIOSPORTING Magazine
PO Box 282, Pine Brook, NJ 07058, USA

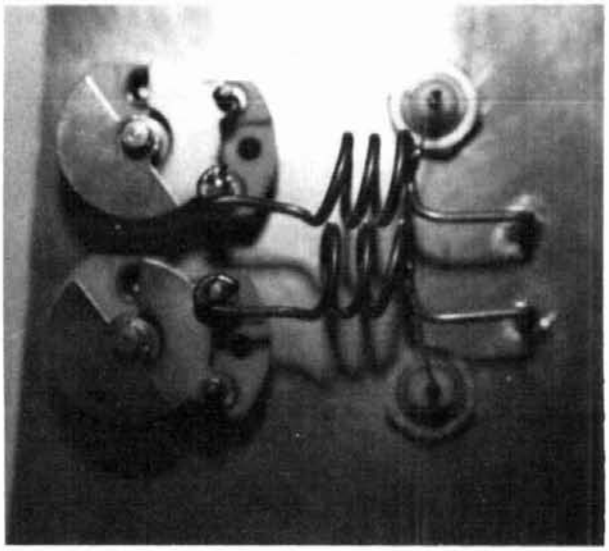


fig. 11. The 220-MHz front-end filter. Air-variable capacitors are necessary for adequate filter Q. The two resonators are closely coupled. The high tap position sets the loaded-Q of the resonators and prevents over-coupling, while maintaining low insertion loss. This filter is exceptionally difficult to tune properly.

reliability — 98 percent of the packets are received without error. This hop is such that 4-dB additional attenuation at one radio caused the packet reliability to be approximately 5 percent, and so was a stringent test of the radios and modems in that the radios were operated near the minimum acceptable received signal level.

The FM-5/K9NG modem experiments gave us valuable insight into proper modem/radio design. We

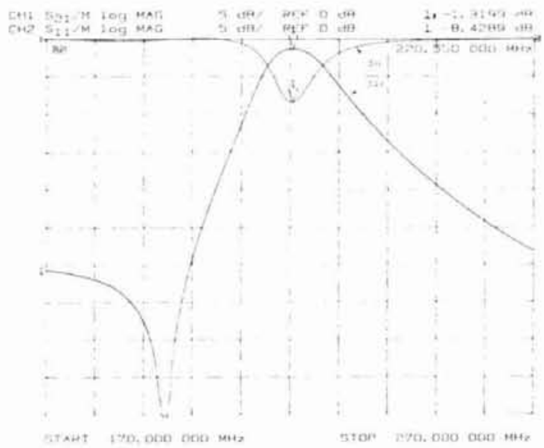


fig. 12. S11 and S12 of the filter. The forward transfer (insertion loss) of the filter is S12, while the return loss (related to input VSWR) is S11. This filter has about 1.3 dB insertion loss, and about 8 dB return loss. The notch is near the image frequency of the 220-MHz receiver.

plan to revise the modem circuits, utilizing surplus commercial 440-MHz equipment for the actual network.

NCP design

The next element in the system is a microprocessor performing as the network switching node. The node control processor (NCP, **figs. 13** and **14**) is an original design, though several modifications have been made since the original artwork was done. The design of the unit is conventional, with a few points emphasized for reliability.

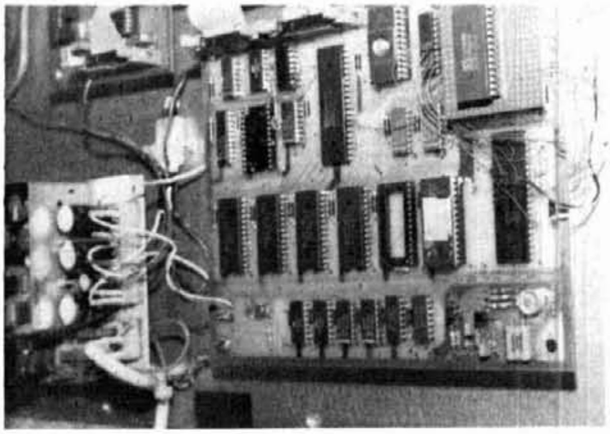


fig. 13. The Node Control Processor (NCP), version 1, with several modifications. A new version 2, which is a multi-channel super-set of the popular TNC-2 unit, is being designed.

The unit contains a Z80 operating at 4 MHz, 32K of EPROM, 32K of RAM, an SIO/2 serial communications IC for the serial HDLC ports, a counter/timer IC (labeled CTC and situated on the modification "ledge" overhanging the main board) for providing the interrupt clock time slices (of 8 ms), and two special circuits. Careful design of the NCP hardware and interrupt daisy chain to match the software was required for the computer to operate reliably at 9600 baud and support multiple HDLC channels simultaneously. All of the I/O devices utilize vectored interrupts through the Z80 interrupt daisy chain. **Figure 15** shows the entire TEXNET node prototype (both radios and modems, ac power supply, and the NCP.

We decided to develop our own board in order to keep costs down and allow the inclusion of two special circuits.

The first circuit is a reliable crystal oscillator. Many logic-gate type of crystal oscillators aren't reliable enough for use in remote, unattended computers. Numerous tests have shown that under certain voltage transients, gate-type oscillators won't start reliably. It's a nuisance to have to climb a tower to recycle the power just to restart a crystal oscillator, and inconvenient to users to have the network out of service during this time. The circuit chosen is a conventional Pierce type, with an additional transistor buffer amplifier. It was tested extensively and found to be robust. (One test to try on an oscillator is to feed the circuit from an adjustable voltage supply. Set the

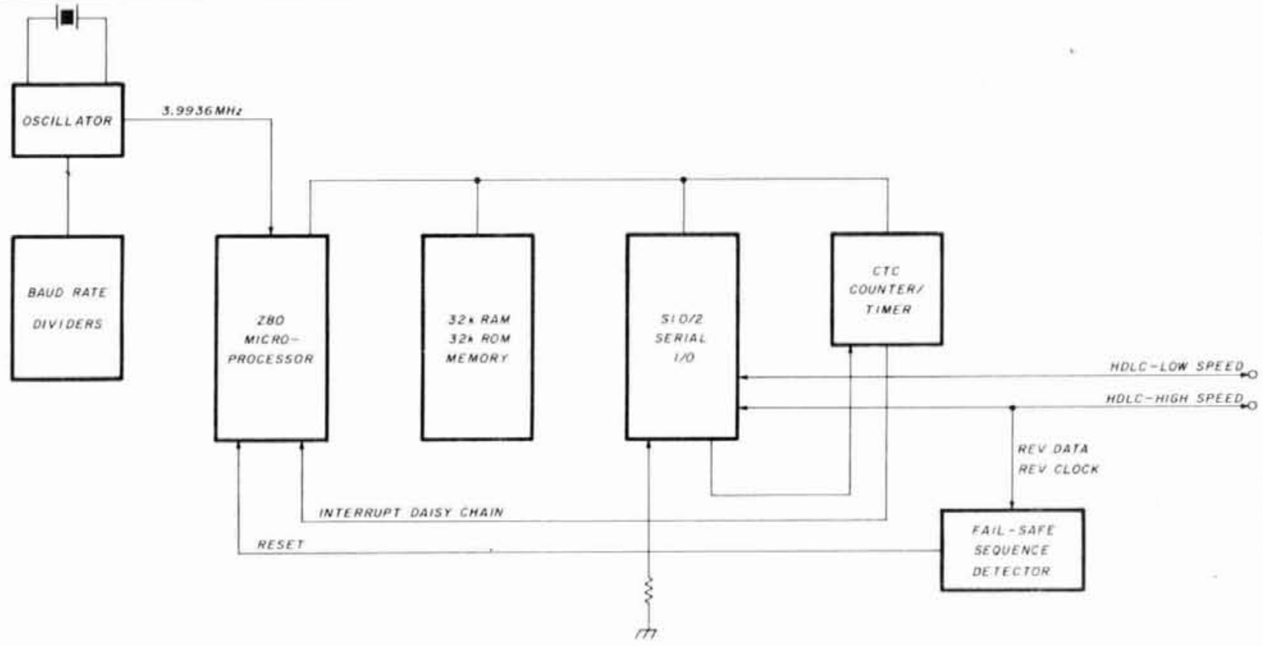


fig. 14. The NCP, showing the crystal oscillator, baud generator, microprocessor, 32 kbytes of EPROM, 32 kbytes of RAM, one HDLC synchronous communication IC, the counter-timer IC, and the fail-safe state machine.

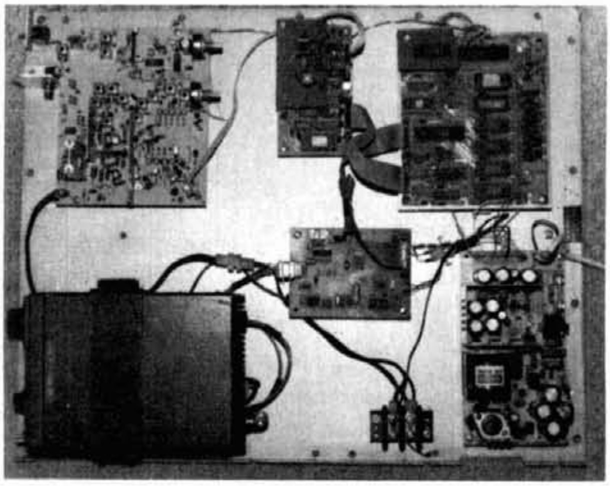


fig. 15. The complete TEXNET node prototype. This is the first prototype of the TEXNET node, which includes the NCP, high-speed modems, the 2-meter radio, the 220-MHz radio, and the ac power supply.

voltage to 0, turn on the power, and very slowly increase the supply potential to the nominal voltage — for example, over a period of 30 seconds. If the oscillator refuses to start, starts on the 3rd harmonic, or starts at the wrong frequency, the circuit should be rejected as unsuitable for unattended operation. Many logic-gate oscillators will fail this test.)

The second special circuit is a fail-safe state machine. This is an EPROM-based logic circuit that monitors the IP data and clock lines (from the high-speed trunk radio), completely independently of the processor or communication ICs. It searches for the presence of a very long (72 bit) sequence. If this sequence is ever detected, the state-machine activates the reset line on the microprocessor, thus restarting the node software. An EPROM contains both the value of the 72-bit sequence and the state-machine code necessary to operate the circuit. Each node is programmed with a different 72-bit sequence, which we've termed the "fire code." Any user of the network can cause the generation of a message with the fire code of a suspected node to be embedded in the message and sent via the network to the suspected node. Thus any node in the network can be rebooted (reset) from any other point within the network. There's a very small chance that ordinary user traffic through the network could resemble the fire code, but since the sequence is so long (72 bits), the mean time between false activation is calculated to be considerably more than 1 million years.

A new version of the NCP will include strappable options, and the circuitry for optional addition of the packet message system (PMS), a 5-megabit hard disk drive-based bulletin board that allows up to ten users to be connected simultaneously. Automatically accessible from anywhere in the network, it's compatible with the W0RLI command set (the most popular packet-radio bulletin board system). **Figure 16** shows the PMS prototype — the world's first turbo-charged TNC-2 clone! This test box contains an ac power supply (vertical), an MFJ-1270 (a TNC-2 clone) with WB5PUC ROM, a disk controller, and a 5-megabit hard drive. The Z80 microprocessor is removed from the TNC-2, and an adapter is plugged in its place. The Z80 plugs into the adapter. The ROM software for both the NCP and the TNC-2 are very similar, except that the NCP supports two radios (or more). The NCP is designed so that it is a superset of the TNC-2 hardware. (Further details will be found Part 3 of this series, which will address software design.)

This concept could be tested as a satellite gateway, perhaps with UOSAT-11, as a store-and-forward message system.

power supply

The power supply for the network node is extremely important in determining the reliability of the network. If the network is to be useful for handling emergency communications, it should be able to survive temporary power outages. Consequently, the TEXNET power supply utilizes a gelled-electrolyte (gel-cel) lead-acid battery, which can provide power for several hours and has a reasonably long life if properly charged and maintained.[3] The power supply for the node utilizes +17 through +24 VDC as the input power source, unregulated (but filtered). The supply/charger (see **fig. 17**) regulates the input to +13.8 VDC through

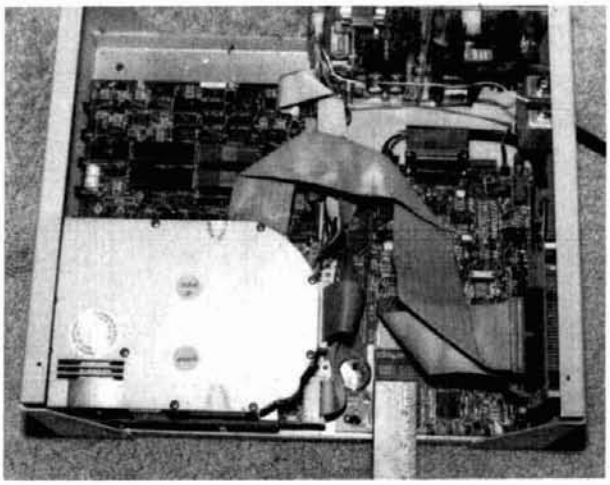


fig. 16. The Packet Message System (PMS), a 5-megabyte hard disk drive-based bulletin board, supports 10 simultaneous users. This prototype includes a TNC-2 clone, the hard drive controller boards, the 5-megabyte hard disk drive, and the ac power supply. A second version is connected to the node prototype, and is accessible from anywhere in the network.

a three-terminal, 10-amp regulator (LM396). The battery charge is controlled by a three-state temperature-compensated charger IC (a Unitrode UC3906). When the ac fails, a relay connects the battery to the load. A capacitor is used to hold up the +5 VDC load voltage for the switching time of the relay. Should the battery completely discharge (i.e., produce less than 12.0 volts dc), the relay protects the battery by disconnecting it from all loads, and the power fails.

The life of a battery is maintained only by very careful charging. This is the reason for the three-state charger. When the battery is initially depleted, it is charged at a constant current of C/10 (at 1/10 the ampere-hour capacity, that's a 10-AH battery charged at a 1-amp current). When the battery voltage rises to nominal voltage (14.25 VDC), a controlled overcharge is initiated. Here the battery is charged at constant voltage (15.00 VDC) until the charge current decreases to C/100 (100 mA for a 10-AH battery). Failure to apply a controlled overcharge will result in the battery's receiving only 80 percent of its previous charge at 14.00 VDC. These voltages are for operation at +25 degrees C and for gel-cels. Liquid electrolyte batteries require different (i.e., lower) voltages. Variations in temperature require compensation in voltage; if compensation is not made, the battery will be severely undercharged at low temperatures and overcharged at high temperatures. The UC3906 contains most of the circuitry, and a temperature-dependent

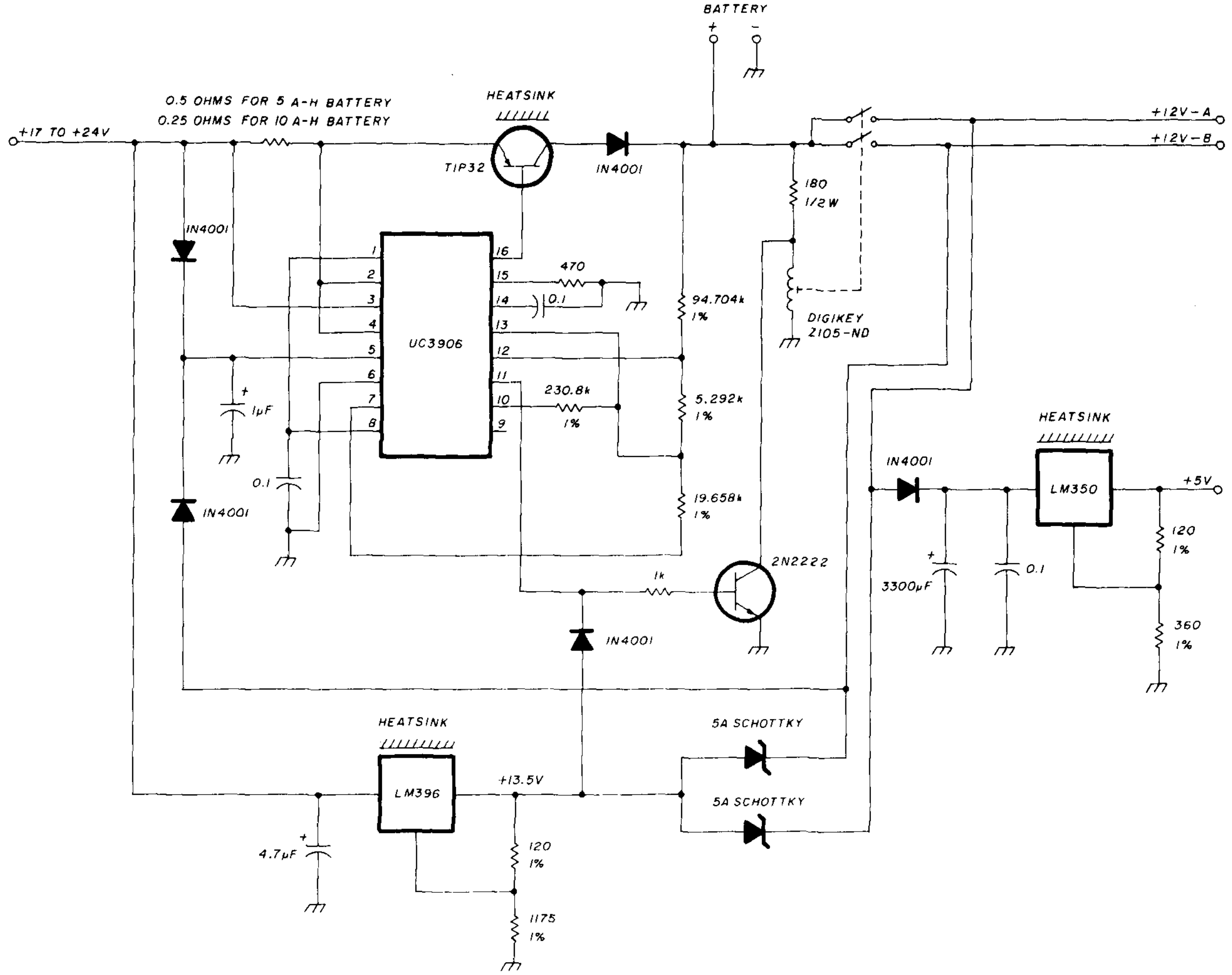


fig. 17. Schematic of battery charger/power supply. The battery charger/power supply is a three-state charger (see text) and power supply regulator. When ac power is present, the battery is charged and the three-terminal regulator supplies power to the node. When ac fails, the battery supplies power to the node. If the battery should completely discharge, the relay will disconnect all loads from the battery, thus protecting it. A large output capacitor on the +12VDC supply to the microprocessor voltage regulator supplies energy for the period of time it takes the relay to energize, preventing a momentary glitch in the +5VDC line.

NEW!

RADIO HANDBOOK 23rd Edition

by Bill Orr W6SAI

Here are some of the highlights of this exciting new edition: New easy-to-use charts for Chebyshev and elliptic filter configuration, new data on power MOSFETS, how to use state-of-the-art OP-AMPS, and home computer RTTY to name just a few examples. New projects include: GaAsFET preamps for 902 and 1296 MHz, easy-to-build audio CW filter, Economy two 3-500Z, 160 meter amplifier, multiband amp using two 3CX800A7's, and a deluxe amplifier with the 3CX1200A7 tube. New antenna projects include: efficient Marconi design for 160 and 80 meters, computer generated dimensions for HF-Yagis, and a 2 meter slot beam. Also all the other information you count on Bill Orr for! You deserve the latest Radio Handbook.
23rd edition ©1986

☐**22424** **Hardbound $26.95**

Reg. 29.95 SAVE $3

Please enclose $3.50 to cover shipping and handling

VISA MasterCard

ham radio magazine BOOKSTORE
GREENVILLE, NH 03048

voltage reference to accomplish this. An external pass-transistor increases the current handling capability to 1 amp. The relay/transistor circuit connects the battery to the load upon ac failure and disconnects the load when the battery is depleted.

Allowing the battery to remain connected to the load after it's depleted will destroy it. New Gel-Cel batteries can normally be expected to operate for six years when properly used; automotive-type batteries, which are designed for high surge currents, will normally have only about a three-year lifetime in a standby power applications and require different voltages. ***Warning: this circuitry is meant only for 12-volt lead-acid gel-cel batteries; its use is not recommended for any other type of batteries. Using Ni-Cad batteries with this charger could result in an explosion.***

references

1. Thomas H. Aschenbrenner, WB5PUC, and Thomas C. McDermott, N5SEG, "The TEXNET Packet-Switching Network — Part 1: System Definition and Design," *ham radio*, March 1987, page 29.
2. Steve Good, K9NG, "Modifying the Hamtronics FM-5 for 9600 BPS Packet Operation," *Fourth ARRL Amateur Radio Computer Network Conference Proceedings*, page 45-51.
3. R. Valley, "IC Provides Optimal Lead-acid Battery Charging Cycles," *EDN*, October 31, 1985, page 163.

ham radio

the HAM STATION

P.O. Box 4405
220 N. Fulton Ave.
Evansville, IN 47710

Store Hours
MON-FRI: 9AM - 6PM
SAT: 9AM - 3PM
CENTRAL TIME

SEND SASE FOR NEW & USED SHEETS
WARRANTY SERVICE CENTER FOR:
ICOM, YAESU, TEN-TEC

TERMS:
Prices Do Not Include Shipping.
Price and Availability Subject to Change Without Notice
Most Orders Shipped The Same Day
COD's Welcome

YAESU

FT-767
- HF/VHF/UHF Base Station
- Plug-in Modules for 6m, 2m, 440 MHz
- Loaded with Features

$ SPECIAL PRICE $

FT-23R with DTMF Keypad
- Mini 2 meter
- 2w or opt. 5w
- 10 memories
- memory and band scan

TEN-TEC
PARAGON
- General Coverage HF Transceiver
  Rec. 100 kHz-29.999 MHz
  TR 1.6 MHz-29.999 MHz
- Microprocessor Controlled Digital PPL Synthesizer, 10Hz Resolution
- 62 High Capacity Memories
- Dual Built-In VFO's
- American Made

ICOM

IC-μ2AT
- Pocket-Sized 2 Meter HT
- TX-140-150 MHz, RC 139-174 MHz
- 10 Memories and a LCD Readout

$ SPECIAL PRICE $

rf concepts
VHF/UHF AMPS
- High VSWR and Overdrive Protection
- 5 Year Warranty, 6 Months on RF Transistors
- All Units have GaAsFET Receive Pre-amps

WELZ
Power Meters
- Large Selection of Meters Always on Hand

$ SPECIAL PRICE $

IC-735
- Most Compact and Advanced Full-Featured HF Transceiver on the Market.

$ SPECIAL PRICE $

DISCOUNTS ON RIGS AND ACCESSORIES FROM: AEA, ARRL, ALINCO, ALLIANCE, ALPHA-DELTA, AMECO, AMERITRON, AMP SUPPLY, ANTENNA SPECIALISTS, ASTRON, BENCHER, BUTTERNUT, B & W, CSI, CALLBOOK, CUSHCRAFT, DAIWA, DIAMOND, ENCOMM, HAL, HEIL, HUSTLER, ICOM, KDK, KANTRONICS, KENPRO, LARSEN, MFJ, MICROLOG, MIRAGE/KLM, NYE, PALOMAR, RF CONCEPTS, ROHN, SANTEC, SHURE, TE SYSTEMS, TELEX/HYGAIN, TEN-TEC, TOKYO HY-POWER, VIBROPLEX, W2AU BALUNS, WELZ, YAESU

For Orders and Price Checks Call 800-523-7731 **Indiana and Information Call 1-812-422-0231**

the TEXNET packet-switching network

part 3: software overview

Z-80 assembly language software offers multi-node, multi-user versatility in real time

In previous installments of this three-part series,[1,2] we described network algorithms and the design and testing of network hardware. This month, we'll discuss node control software.

Contained in a single 27C256 ROM, the software determines the functions and user features offered by the network nodes. Highly modular in design, the software was written in Z-80 assembly language for two reasons: one, because the wide array of services offered by a single node put memory space at a premium; and two, because one or more of each node's ports would be operating at 9600 bps, therefore stressing real-time capacity.

Because space is limited, we'll discuss the functions of specific software areas rather than describe the software itself in detail. **Figure 1** illustrates a typical node, with each major software area indicated by a circled reference letter.

common logic

The common logic portion of the software package, identified as *A* in **fig. 1**, is responsible for memory allocation and management, real-time scheduling, and interfacing to the various hardware I/O devices, such as SIO's and the CTC. As an example of these housekeeping tasks, let's look at memory management. Because there are many more users for memory than there is memory capacity, memory management — specifically in regard to time-sharing — is critical to efficient system operation. The generic problem with most memory management schemes, however, is *deadlock*, which occurs, for example, when a memory user has some memory allocated and needs more to complete the job, but can't get more because of what's already been assigned. The node software package follows a procedure known as *load shedding* to prevent deadlock; it does this by finding the "oldest" and largest consumer of memory and aborting his resource allocation.

The largest section of the common logic is the multi virtual connection PAD (Packet Assembly/Disassembly) logic. This general-purpose software has a standard interface to the higher layers of software wishing to use its services. The PAD is completely state table driven and currently implements the ARRL AX.25 V1.0 and V2.0 protocol specification. The PAD supports a variable number of simultaneous virtual connections

Thomas H. Aschenbrenner, WB5PUC, and Thomas C. McDermott, N5EG, Texas Packet Radio Society, P.O. Box 831566, Richardson, Texas 75083-1566

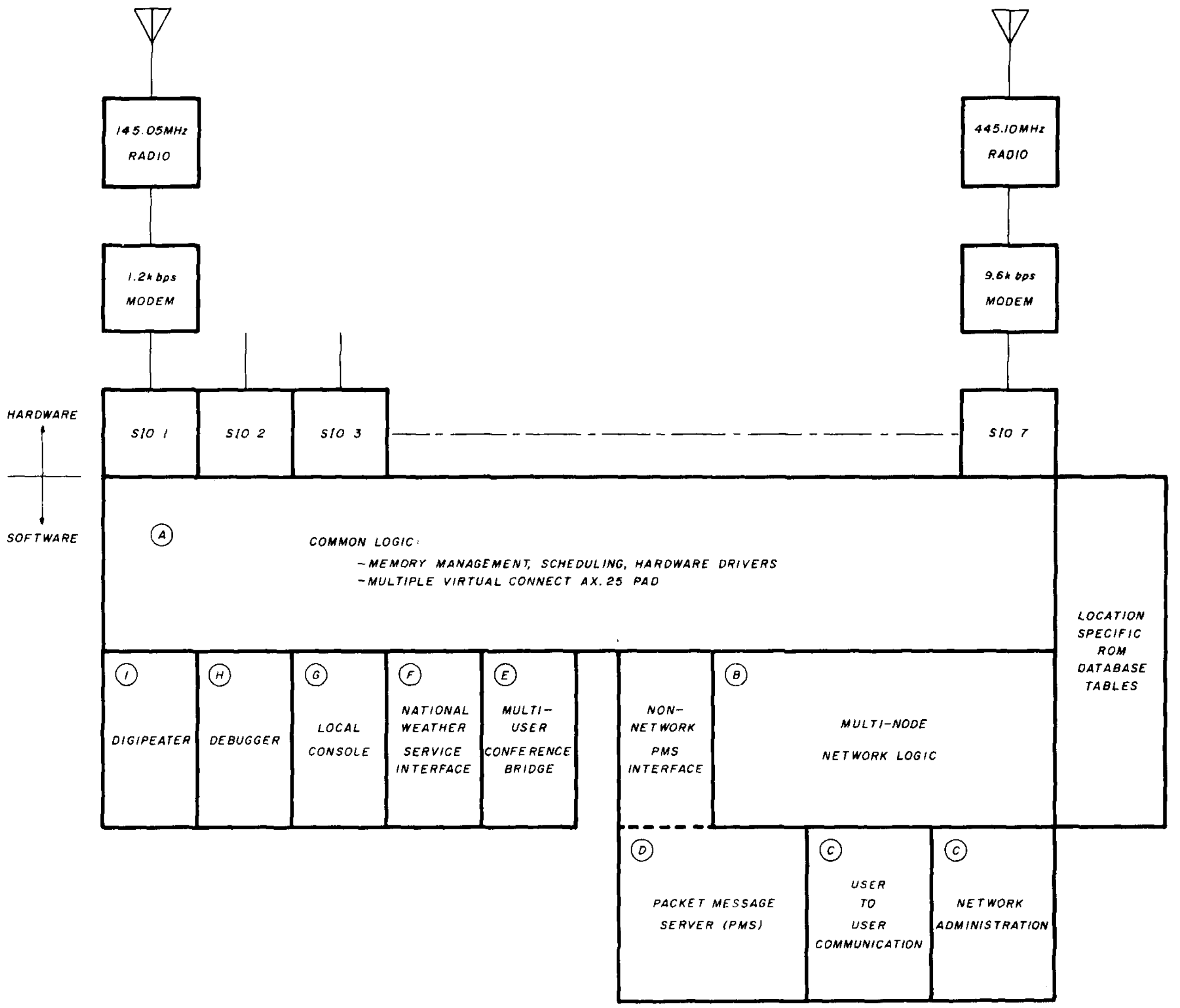


fig. 1. Node software block diagram shows hardware/software boundary and layering of various software functions.

(currently set to 20), with any of the virtual connections conforming to either V1.0 or V2.0 of AX.25. Selection of V1 or V2 operation is determined by the users' setting of their TNCs — for example, if they're running V1, then the node will act like a V1 TNC when they connect. Users running V2 will see the node as a V2 device.

The PAD, which supports up to eight physical communications channels (four SIO's), has been tested in a multiport configuration with both 9600 bps and 1200 bps, operating simultaneously without loss of data. In order to achieve this performance level for 9600-bps operation, a nested interrupt structure (the interrupt service routines are themselves interruptible) which would allow timely response to interrupts from the 9600-bps port(s) had to be designed.

The common logic provides a number of features of the node. All users of the node see the node as a series of AX.25 addresses. For example, this is how users see the Garland, Texas node:

W9DDD-2	**1st conference bridge**
W9DDD-3	**2nd conference bridge**
W9DDD-4	**TEXNET Access**
W9DDD-5	**Node's local console**
W9DDD-6	**Test access**

These applications will be explained shortly; what's important to note here is that all of the addresses have the same Amateur call, W9DDD, and that the application is selected according to the SSID. (This method of operation is only one configuration of the address database. Instead of using the same call (W9DDD) five times, different calls could have been configured, with the SSID held constant. The node will support any combination of the above examples.)

The common logic also supports various restriction and parameter tables. New connections to any AX.25 address of the node can be inhibited as a function of the number of digipeaters used to get to the node. This has been found to be useful in reducing channel congestion attributable to excessive retries on long digipeater paths to a specific node. The preferred method is to put a network node near enough to the user group and carry the traffic on the network backbone trunks.

Parameters managed by the common logic, on a per-physical port basis, include all of those specified in the AX.25 protocol (i.e., T1, T2, T3, K, N2, etc.) and some unique to this application. It was found desirable to define the AX.25 T3 timer as either an all-seems-well timer (its original function) or an auto-disconnect timer. In the auto-disconnect mode, if a user's virtual connection is idle for greater than the T3 time value (nominally 3 minutes) he or she is automatically disconnected from the node, thereby making room for other users. This mode can be overridden by the ALERT network mode, which will be discussed below.

Finally, the common logic is responsible for gathering statistics for the node. Two main groups of statistics are collected. The first are those that aid in "traffic engineering" the node. Quantities such as the amount of memory in use, the maximum amount ever used, and the total available allow visibility into the level of service being provided and indicate whether or not more RAM should be allocated to the free memory pool, thus decreasing memory space available for applications. Experience indicates that a free memory pool of approximately 30K, allocatable in about 200-byte chunks, provides good service with enough reserve capacity to handle rather large impulse loads, such as congestion on 9600-bps trunk circuits.

The second group of statistics collected are those having to do with node use. Quantities such as the number of frames transmitted, received, and retransmitted for each physical channel yield data on overall network use, thereby suggesting possible additions to the node or changes in network configuration. These numbers can also be used to determine the performance level of network trunks.

multi-node network logic

As illustrated, the multi-node network logic (see *B* in **fig. 1**) is supported by the common logic. In turn, it supports higher-level applications such as Network Administration, User Intercommunication, and the Packet Message Server.

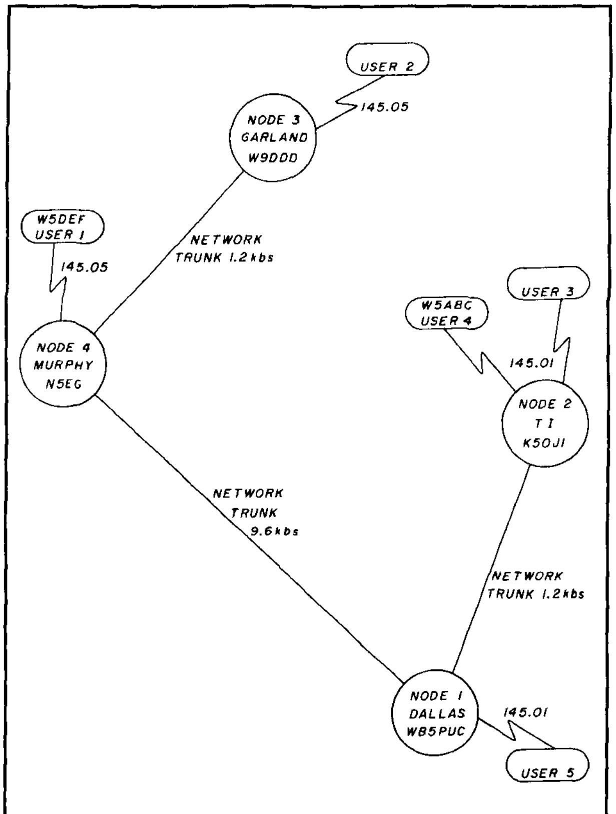


fig. 2. Dallas network test configuration.

The multi-node network logic is a datagram-based system that can support up to 256 node locations with as many as 20 simultaneous users at each node. All network nodes interconnect via permanent virtual connections between them.

The network is a database-driven system that features an extremely user-friendly termination-based routing structure. The network provides end-to-end flow control to eliminate internal congestion. A user's TNC "going busy" causes a network message to be sent to the far node, which in turn will "busy" the subject port at the far node, thus causing the remote user's TNC to stop sending.

In order to allow for an increased level of reliability, the network allows for alternate routing of data via multiple routes to a single node. Controlled by the node's database, alternate routing is automatically performed upon detection by a node that its first-choice route has failed.

The network provides substantial feedback to the user community via a mechanism called *Network Information Codes* (NIC). These NICs are printed at a user's terminal when something unusual happens that will affect the performance of the network from the user's viewpoint.

For an example of NIC operation, and for further illustration and explanation, see **fig. 2**, which illustrates the network test configuration as it was in Dallas when this article was being prepared. This test configuration was used for software testing during the development phase and for a series of operational tests during the beta-test phase.

The system shown in **fig. 2** is located in the north Dallas area. No particular attention was paid to the geographical locations except for convenience of access for testing. The illustrated network architecture was chosen because it provides for testing of all possible configurations of actual network operation.

Each of the four nodes has an Amateur call sign assigned as its user-access AX.25 address. Each node also has a mnemonic name to which all users refer when asking the network for services. For example, Node 4 has a user-access address of N5EG-4, and is referred to by users in all network commands as MURPHY. (Note: MURPHY is named for its location in Murphy, Texas — not in honor of the universal law of the same name.)

In an actual geographically dispersed network, the user community around each node has to know only their own node's user access address (N5EG-4 in the example above). They refer to all other nodes in the network by the network node names (GARLAND, *MURPHY, DALLAS, TI [Texas Instruments Radio* Club] — see **fig. 2**).

Also shown in **fig. 2** are typical user stations, labeled User I through User 5. These stations are standard Amateur packet stations equipped with commercially available TNCs and VHF transceivers.

the network users' interface

The Network Users' Interface (see *C* in **fig. 1**) provides the user's view of the network. Referring to **fig. 2**, if User 5 were to connect to WB5PUC-4, he would see the following displayed on his screen:

```
C WB5PUC-4
***Connected to WB5PUC-4
WB5PUC-4 Virtual Connection 06 at 18:32:20 on 11/20/86
***Welcome to TEXNET***
Network Cmd?
```

At this point he can issue any of the network commands or just disconnect if he's finished.

network command structure

The network command structure is as follows:

- **@NODENAME**. The NODENAME field may consist of any valid network node name. Valid node names may be two to seven characters long and can consist of any ASCII character except carriage return. As indicated in **fig. 2**, these names are MURPHY, GARLAND, TI, and DALLAS in our test configuration. User entry of a name not recognized by the node as valid will result in a message to the user indicating that an invalid node name has been entered. A list of valid names will be printed to allow correction of the error.

Because any command may be destined for any node, most commands are terminated by the @ NODENAME field. Some commands have an implied node name. Commands that are currently implemented are listed in **table 1**. For purposes of this description, the network commands are divided into two categories: User and Administration. Note that this division is for explanation only; any user may execute any command. Those who simply want to communicate via the network need learn only the commands listed in the User category. A much smaller group of people — those who are responsible for network engineering and administration — need to learn the Administration commands.

While the command words are spelled out fully in **table 1**, only the first character must be entered for most commands — for example, ***H*** for *HELP*. **Table 1** includes examples of network commands and their abbreviated formats.

- **HELP**. The user entering this command is given a partial list of commands (those marked "user") and referred to the network operation manual for further enlightenment. This strategy, rather than the on-line tutorial method, was chosen in order to reduce channel congestion. All new users are sent a copy of the TEXNET manual; this eliminates trial-and-error learn-

Table 1. Active network commands can be typed by user in response to the network command prompt, which is received after the user does a standard connect to the node.

Type	Command	Parameters () = optional input	Example using abbreviated command
User	Help	None	H
User	Circuit	Call sign (via Digi)	C W5ABC @ Dallas C W5ABC V WA5LXS @ Dallas
User	Locations	None	L
User	Message	None	M
User	Alert-on	None	A -on
User	Alert-off	None	A -off
Admin	Statistics	None	S @ Dallas
Admin	Initialize	None	I @ Dallas
Admin	Time	(DDMMYYHHMM)	T @ Dallas T 0108871420 @ Dallas
Admin	Route add	26 bytes of information	R A 01 . . . . 7F @ Dallas
Admin	Route delete	Number	R D 2 @ Dallas
Admin	Point	Function number	P E 2 @ Dallas P D 2 @ Dallas P S @ Dallas

ing on the air, which results in more efficient use of channel space.

- **The CIRCUIT** command tells the network the desired destination of the user's connection. See **fig. 2**; if User I wanted to communicate with User 4, he would enter the following at the command prompt:

C W5ABC @ TI

Obviously, this example assumes that User 4's call sign is W5ABC. Note that User 1 doesn't need to know how the network will connect to User 4; he just enters the terminating node name TI, and it's the network's responsibility to figure out how to route the message to TI.

The routing strategy is accomplished by the database in a node's knowing which of its adjacent nodes should be used to get to a remote node. It's a tradeoff, in that it makes the node database somewhat more complex — but it does allow the users an extremely easy interface.

Because the users don't need to know the physical configuration of the network in order to communicate, the network administration group can change it at will without having to inform (and thus re-educate) the entire user community. This routing strategy takes advantage of the disparity between the number of times users access the network to communicate (very large) versus the number of times the administration group adds or makes changes to a node (very small). The increased burden on the administration group is more than offset by the benefit to users.

In order to compensate for incomplete geographic network coverage, the CIRCUIT command allows an optional string of digipeaters to try to connect to the desired station. Thus,

C W5ABC V WA5LXS @ TI

would cause the remote node TI to attempt to connect to W5ABC, using WA5LXS as a digipeater. Up to two digipeaters can be specified in the CIRCUIT command.

Operations note: whenever a remote node attempts to connect to a station (as when TI attempts to connect to W5ABC in our example), the node will use Version 2 of the AX.25 protocol unless a digipeater is specified. If a digipeater is specified, the remote node will attempt the connect using Version 1 AX.25. This is because there are still some digipeaters around that won't accommodate Version 2.

After User I enters the CIRCUIT network command, one of three things will happen. First, his CRT may display the message,

Your connection is established

in which case he's being advised that the desired remote user (User 4) is on line. At the far side, our remote user (User 4) would receive the following on his screen:

*****Connected to K5OJI-4**
*****Linked to W5DEF at Murphy via Texnet**

Therefore, User 4 knows exactly to whom he is connected via the network, and where the originating station is located. The above example, of course, assumes User 1's call is W5DEF.

The "**** Linked to" string received from the network allows a W0RLI-compatible BBS system to know who the real user is (W5DEF) rather than thinking it is connected to the node (K5OJI-4).

The second possible message is:

Remote user not responding

In this case, User 1 is informed that connection is impossible. This could occur for a number of reasons: the remote station may not have its equipment turned on, or it may be turned on but involved in another QSO.

The third possibility is receipt of an NIC message. Using our example, if user 1 received:

Network information code 017 from Dallas

he could look in the TEXNET manual and find that code 017 means his attempt was routed as far as Dallas, but couldn't be routed further because of a network trunk outage. This can then be reported to network administration for remedial action.

- The **LOCATIONS** command allows users to ask the node for a list of remote locations which can be reached through the network.

- **The MESSAGE** *command gives any network user,* regardless of node location, access to the network Packet Message Server (PMS) logic. Details of the PMS subsystem will be covered below; for the moment, let's just say that it's a network-wide message file system similar to the W0RLI bulletin board system.

It's important to note that users at any node in the network don't need to know where in the network the PMS system is physically located. All a user needs to do is type MESSAGE or M, and the network takes care of routing to PMS. In the test configuration shown in **fig. 2**, PMS is actually located at Node 1 or Dallas, but very few users are aware of this, since they can connect to any node to access PMS. Once the network has established a connection for the user to PMS, the user can enter PMS commands to store, list, or read messages.

- **ALERT-ON** AND **ALERT-OFF** enable or disable a special mode of operation called ALERT, which is especially designed for accommodating emergency traffic handling via packet radio. The ALERT mode can be enabled from any node in the network. When a user connects to any node and issues the ALERT-ON network command, his or her node will send a "broadcast" command to all other nodes in the network informing them that ALERT mode is being enabled and telling each the name of the originating node. At this point, several things happen in every node.

Users connecting to the network are informed that an ALERT is in progress. Let's assume that a user at Dallas has enabled the ALERT mode. As in our previous example, User 1 wants to communicate with User 4; when he tries to connect, however, he receives the following message on his CRT:

```
***Connected to N5EG-4
N5EG-4 Virtual Connection 03 at 08:30:20 on 12/15/86
Pls disconnect unless your traffic is related to the network
alert in progress from Dallas.
**Welcome to TEXNET**
Network cmd?
```

When ALERT is enabled, all user automatic disconnect timing is disabled. Thus, instead of the standard 3-minute idle time disconnect, to which all users are subject, all nodes will allow connections of unlimited duration to their ports. This disconnect timing suspension affects all connects to the node. Thus, if groups of users want to use their node's conference bridges to handle emergency traffic, they can remain connected indefinitely. User connects to PMS may also be of indefinite duration.

With the enabling of ALERT, a special mode of PMS that provides a real-time message exchange between the multiple users connected to PMS is also enabled. Thus, when one user SENDS (see PMS description below) to another, all of the standard PMS functions are invoked. In addition, after automatically saving the message on disk, PMS will check to see if the addressee is currently connected to PMS on another of its logical ports. If he is, PMS will automatically display the message at the addressee's terminal.

With this mode of operation enabled, PMS becomes a real-time message forwarding system among its connected users, with the added feature that all messages are archived to the disk. This feature can be extremely useful in emergency government communications back-up, since the stations connected to PMS could be physically located anywhere along the network.

For a Department of Public Safety (DPS) exercise, for example, Amateurs equipped with standard packet equipment could be located in each community's DPS office. Each station would be connected, via the network, to one of the logical ports of PMS. Because the ALERT mode would be enabled, they would be able to stay connected indefinitely — remember, DISC (disconnect) timing is inhibited with the enabling of ALERT — and each time one station uses the standard **S** feature of PMS, the message would be displayed in real time at the receive station. Of course, the message would be automatically saved to disk so the receive station can review it at will. Other advantages of this technique include a complete on-disk record of all messages (useful for exercise postmortems) and the fact that other connected stations may review all communications except those sent as private messages.

- **The STATISTICS** command, when issued with the NODENAME parameter, causes the local node to acquire the operational statistics of the remote nodes. The statistics counters in a node aren't cleared by this command; this allows timed interval measurements to be taken. Every midnight, all statistics counters are cleared to zero.

The following are the statistics kept at each node and therefore available via the STATISTICS command:

Frame buffers available
Frame buffers in use
Maximum frame buffers ever used
Total connects
Connects to weather
Connects to conference bridge
Connects to network
Network circuits active
Maximum network circuits ever active
Packets sent on each physical channel
Packets received on each physical channel
Packets re-sent on each physical channel

In addition, the real time at the subject node is sent back with the statistics. By looking at some of the statistics returned, the network administrators can make various engineering judgments about the level of service being provided by a node to its user community. Quantitative measurements of the activity of a node's local user community, and of which node services are being used, can also be made.

- **The INITIALIZE** command is the means of remotely restarting any node. When a node receives an INITIALIZE command directed to it from someplace on the network, two functions are executed. First, upon receipt and decoding of the command, the software kills all activity in the node for 30 seconds. This delay allows time for adjacent nodes to have their network trunks to the subject node time out because of the subject nodes' lack of activity. This action gracefully removes the subject node from the network fabric. At the end of the delay period, the subject node does a cold restart, thus appearing to the rest of the network as if it had just been turned on. In response to this action, the subject node — after consulting its database — establishes network trunks to the appropriate adjacent nodes. Network operation is now re-established.

The second thing that happens upon receipt of an INITIALIZE command is the activation of an external hardware fail-safe circuit. Buried in the INITIALIZE command as it traverses the network is a unique bit sequence generated by the originating node and specific to the subject node. Upon detection of the bit sequence by the subject node's external hardware fail-safe circuit, the master reset line of the Z-80 is asserted. This technique obviates the above discussion on the software execution of the INITIALIZE command unless, of course, there's a failure in the fail-safe hardware itself. The combination of the two techniques would require a double failure in a node before the ability to remotely reset it would be lost.

- **The TIME** command, when issued by a user at any node and directed to a specific remote node, will cause the real-time clock at the remote node to be updated to the time contained in the message. If no time parameter is input by the originating user, then the current time at the user's node is sent to the remote. If the time parameter is entered by the user, his or her node's real-time clock is updated with the input time before its value is sent to the remote node.

- **ROUTE ADD/DELETE**. Since in every node the ROM routing table is copied to RAM for operation, it may be changed by being added to or deleted from. The ROUTE commands are the means by which new nodes can temporarily be added to an existing network or by which the network configuration can be changed to accommodate a failure or some other special event.

- **POINT** COMMAND. This command is used to control external equipment at any node site. Within each node control point (NCP), there are 5 bits of input and 5 bits of output available for external use. These bits, called control points, could be used to control and monitor anything that interfaces via contact closures. Co-located equipment at the node site, such as other repeaters, could take full advantage of digital control via this feature of the NCP.

The POINT command allows full on/off control of the output points. For example, suppose a co-located voice repeater at the GARLAND node needed to be controlled. NCP output Point 1 could then be wired to the voice repeater's control relay, and perhaps NCP input Point 1 would be wired to one of the control relay contacts to allow monitoring of relay closure.

Any authorized packet station, anywhere on the network, can issue the following command to enable the co-located voice repeater:

POINT ENABLE 1 @ GARLAND

or for short,

P E 1 @ GARLAND

Displayed on the CRT — after the network has passed the POINT command to the GARLAND node and the command was executed — would be:

```
Control Points at Garland Are
Point:    1 2 3 4 5 6 7 8
Input:    E D D D D D D D
Output:   E   D D D D D D
```

Since the control operator knows that Control Point 1 is wired to the voice repeater, he can see that it's enabled, and by looking at Input Point 1, confirm that the control relay is closed.

Any time the control operator wishes to check if the voice repeater is on, he needs only to type **P S @ GARLAND** to get the status display of the control functions. When he chooses to shut the voice repeater down, he enters **P D 1 @ GARLAND** at the network command prompt. Again the status will be displayed on his screen in response to the command, allowing confirmation that shutdown has occurred.

One important use of the POINT command is the control of a pair of control points wired over to the node's Uninterruptible Power Supply (UPS). By design, the UPS has a control lead which, when enabled, forces the UPS to switch from ac to battery. Another UPS lead provides an indication that the UPS has switched to battery operation.

In a normal node configuration, Control Point 5 input and output are wired to the UPS control leads. This allows any node in the network to be instructed to operate off battery power by the simple issuance of the **P E 5 @ Node** command. Issuing the **P D 5 @ Node** command restores the node to ac operation. This feature allows weekly testing of the UPS to ensure that it would be effective during an emergency.

network internals

It may be interesting at this point to describe some of the internal workings of the network software, which has the ability to establish, kill, and communicate over any number of virtual connections. Thus, at system startup, the network application executes logic to find out who its neighboring nodes are. It then establishes a virtual connection to each, over whatever physical channel is specified to be used as the network trunk. This virtual connection is left up forever. All subsequent communications from this node to its neighbor, whether user data or network management data, travel over this permanent virtual connection. Additional logic determines the network configuration for the node's routing table.

A special byte string is added to the beginning of all packets going over a network trunk. This string, known as a *Network Header Block* (NHB), consists of a minimum of 5 bytes:

NHB.RNN	**Destination Node Number**
NHB.RLC	**Destination Virtual Connection Number**
NHB.LNN	**Originating Node Number**
NHB.LLC	**Originating Virtual Connection Number**
NHB.NCF	**Network Control Field**
~ ~	**Any Network or User Data**

When a node receives information from a virtual connection marked as a network trunk, it examines the NHB. Looking at NHB element **NHB.RNN**, it checks to see if the received string is for this node. If it isn't, the node consults its routing table to see on which of its trunk circuits (virtual connections) it is to retransmit the string. This is known as *transit routing*, and it's extremely fast, thereby yielding a large node bandwidth in this mode. If the examination of **NHB.RNN** confirms that the string is for this node, the **NHB.NCF** byte is decoded to tell the node the proper action to take on the received string. There are over 15 valid network control fields in the current design, so it wouldn't be practical to cover them all here; therefore, we'll choose just one as a simple example.

The example will be of a remote node (perhaps many hops, or nodes, away — we don't know, and we don't care if the information came to us via transit routing through multiple nodes) asking us to send our operational statistics. When we finally receive the string, we find it's directed to our node; moreover, after examining the network control field, we see it is set at 01 HEX, which tells us that the remote wants us to send our collected statistics. Our response to this is to reverse the NHB items so we can send information back to the requester and reset the network control field to 02 HEX, which will inform the requester, when he receives the string, that we responded. We append approximately 40 bytes to the string. These appended bytes are the operational statistics we've been collecting, which is what the remote requested.

the packet message server

The PMS logic (see *D* in **fig. 1**), resident within a node, allows a simple three-chip interface to a Western Digital WD 1002-05G disk controller and a standard ST506 5-Megabyte hard disk to become a network wide integrated message storage/retrieval system. Normally, only one disk is required per network. It's possible, however, in a very large network, to assign specific groups of nodes to a given PMS, in which case there could be more than one PMS disk. The PMS logic is resident in each node, but only the node(s) equipped with a physical disk allow it to execute.

From an individual user's viewpoint, the PMS looks and acts like a W0RLI bulletin board system. This choice of operational methods was made to reduce the amount of end-user training required. It also aids in the transition that must occur when all users in a given area are switching from being served by an in-place W0RLI system to the PMS system running on the network. Because of the similarities between the familiar W0RLI system and the PMS, this discussion will not cover details such as how the SEND, READ, KILL, and other commands work, but will instead concentrate on the PMS's enhancements.

Because the PMS system is designed to provide message service for all the users of a network subregion, the bandwidth requirements are greater than those found in other message systems. Unlike existing message systems, which have the ability to service only one user at a time, the PMS system allows up to ten users to log on simultaneously, providing the same grade of service, relative to response time, to each.

The storage element in a PMS is a 5-Megabyte hard disk. These disks are equipped with four head assemblies and a platter assembly capable of accommodating 154 cylinders. To make use of these characteristics and to provide the response time needed, the software was expressly designed to have a message file structure that takes advantage of the physical characteristics of the disk. The first time a new node is energized, the disk is automatically formatted and then configured to the necessary file structure. Every restart thereafter preserves all saved messages.

The combination of the hard drive and special file structure results in an incredibly small response time; even with multiple users, each user has a real-time response from the PMS, with a delay of less than 1 second. In tests run on the Dallas network test configuration, remote network users accessing the PMS via a 9600-bps network trunk observed no difference in response time from that observed by users directly connected to a PMS system. Both experienced a response time of less than 1 second. Most operational response time delay was attributable to congestion on the 1200-bps final link to the user.

The PMS system supports up to 500 active messages from a message number range of 1 to 99,999. There's no difference in response time if a user is accessing Message No. 1, 500, or 50,000.

All active messages are subject to the auto-delete function of the PMS. An undeleted message will remain in PMS for no fewer than 14 days and no more than 28 days before being automatically deleted by the system. While these times are variable, users seem to find them satisfactory.

Important note: the PMS message system is designed to run *completely* unattended. No SYSOP is required. All duties previously performed by people running mailboxes are done automatically by the PMS logic.

The PMS system can run "stand-alone" — for example, with its user community geographically near the PMS equipment, thus replacing an existing mailbox system. In testing this configuration, a version of the software was put into an MFJ 1270 TNC-2 with a hard disk interfaced to it. This system is currently used as a demo system for other groups of Amateurs wishing to participate in TEXNET.

PMS is used primarily as a network-wide message system. In its network configuration, any user located anywhere on the network may, after receiving the network command prompt, issue only the MESSAGE command to be automatically routed over the network to his servicing PMS. Referring again to **fig. 2**, the PMS equipment (all that's required in addition to an existing NCP is the disk controller and hard drive) is physically located at Node 1 in our Dallas test system. Any user can connect to any other node by issuing a connect request to the desired nodes call and using the -4 SSID. After this, all that must be done is to issue the MESSAGE command. The next thing that appears on the CRT is the text from PMS.

Since the PMS is network-compatible, it knows who the originator is as well as where (i.e., at what node name — DALLAS, GARLAND, TI, etc.) he's located. All of this is used automatically whenever a user does an **S** command to send a message to another user. If, after doing the message send, the user does a simple **R** (in PMS, the **R** with no qualifier or number will read back the last message number), he'll see that his call and the name of his node have been automatically entered in the message header.

In order to provide a message interface between an entire network and the existing Amateur message forwarding system, the PMS supports a subset of message forwarding. All messages that network users enter into PMS and which require forwarding will be passed by PMS to a single W0RLI system for eventual forwarding by the existing systems. The identification of this W0RLI system is contained in the node's database. This same W0RLI system can also forward messages into the PMS system for reading by all network users. PMS is designed to follow the standard forwarding protocol and uses a direct access port to connect with the W0RLI system. For example, in **fig. 2**, Node 1, containing the PMS system, will access the WA5MWD BBS system in Dallas by using its WB5PUC-7 direct access port. The WA5MWD BBS system can pass messages into the network by connecting to WB5PUC-7. The WA5MWD box doesn't know it's talking to the network system; it thinks it's talking to just another standard W0RLI system.

non-network local services

The following are services provided by each network node on a standard basis. These services are independent of the network, but still extremely useful:

Multi-user Conference Bridge
National Weather Service interface
Local Node Console
Debug Aid
Digipeating

multi-user conference bridge[3]

The multi-user conference bridge (see *E* in **fig. 1**) logic allows up to six remote users to hold a roundtable conversation with each remote, with the ability *to see all text generated by all other remotes*. Each remote user has a direct AX.25 connect to a logical port on the conference bridge. Upon reception of a packet of information from one user, the bridge will make multiple copies and send one to each of the other connected users. Since each user has an AX.25 connect to the bridge, he's assured of not losing packets, since the bridge will retry upon lack of an acknowledgment from the affected remote.

As the text is being transmitted to a remote user, it's modified to show which of the other remotes originated it. Therefore, all users not only see all text from each other, but know who originated it.

In a standard software package, the conference bridge logic simultaneously supports two completely independent six-party conferences. Each of these independent conference bridges is accessed by remotes using unique SSIDs, (usually -2 and -3).

Since the bridge logic is supported by the common logic, any of the remote users may be operating in either AX.25 version 1 or 2. Text is transferred among users without regard to versions used by individual remotes.

At any time during a conference, any of the users may type **CONTROL-U**. Upon receipt of this character, the conference bridge logic will respond by sending the requester a list of call signs of all other remote users currently connected to the conference bridge. Any of the remotes may exit an established conference, and other remotes may join (on a noninterference basis) the conference in progress.

Tests conducted in the Dallas area show the conference bridge to be a cleaner and more reliable way for groups to hold multi-user roundtable connects than the UNPROTO mode available on standard TNCs. This is because of the built-in advantage of error-free AX.25 connects combined with the fact that each remote has to have a good path only to the conference bridge, not to all other users.

NWS interface

The National Weather Service (NWS) application (see *F* in **fig. 1**) runs concurrently with others at a node. Typically, one node in each region could have this application enabled and interfaced via a standard 75-wpm, Baudot-encoded, 20-mA landline to the National Weather Service.

The NWS wire feed provides raw weather data for a large geographic region. The NWS logic monitors all received data, but selects and stores only those NWS products (for example, region forecasts, severe storm alerts, thunderstorm warnings, etc.) which have their unique codes programmed into the node's database. The NWS logic currently supports 30 code sequences, with each having the ability to be from 2 to 11 characters in length. This code format is consistent with the nine-character sequence utilized as a standard by the NWS.

Remote users connect to the node's NWS logic by means of a unique SSID (usually -1). Up to ten remote users may be connected to a node's NWS logic simultaneously. Upon connection to the logic, the system will wait for the remote user to enter a single product designator — for example, a user in Dallas who enters "D" causes the node to send the current Dallas area weather forecast. Entering a question mark prints the entire list of stored product designators.

All remote users are assured of receiving the latest data because the node updates its buffers in real time, as new information is received from the weather bureau. At 2 A.M. each day, the node clears all buffers to eliminate products sent infrequently by the weather service.

local CRT console

Figure 1 (see *G*) shows the local console CRT logic "sitting on top of" the common logic. The CRT logic at each node allows a locally connected standard ASCII CRT to originate and terminate connects with any standard TNC. In operation, it uses a unique SSID (usually -5) to distinguish itself from other node services. The local console logic isn't meant to be a full TNC-human user interface, as commercial TNCs are. Instead, this logic provides a minimum subset of human user commands that are necessary for testing and administration of the node. **Table 2** contains examples of command types supported; some specific commands follow.

Table 2. Command types supported by TEXNET's CRT logic.

Command Type	Format
ORIGINATE ON PHYSICAL CHANNEL	ON where N = 0, 1, etc. = physical channel
CONNECT	C W5ABC V WA5MWD, N5EG
DISCONNECT	D
BUSY	B
FRAME TRACE	FT
LOCAL TIME SET	LT DDMMYYHHMM
STANDARD MODE	SM
VERSION SELECT	V1

- **The ORIGINATE ON PHYSICAL CHANNEL command** allows the user to select which physical channel is to be used for originating connects. The console will accept connects from any physical channel, but will do so only if it's not currently busy. This command allows the console to originate on the physical channel connected to the 1200-bps radios and, therefore, look like a standard user. Also allowed are connections to be established to any other node site via the 9600-bps trunk circuits. The latter is particularly useful for troubleshooting remote nodes from our network control site.

- **CONNECT AND DISCONNECT** are identical to the commands on any standard TNC.

- **BUSY** sets the console into a busy state to facilitate troubleshooting the network's end-to-end flow control logic.

- **FRAME TRACE** is similar to that command on any standard TNC in that it provides a real-time look at all frames as they're received by any physical channel on the node. Displayed are both hex and ASCII equivalents of the received frame.

- **LOCAL TIME SET** allows maintenance personnel to reset the real-time wall clock at a node. This command usually isn't used because it's possible to set the time at any node in the network from any other node using the network TIME command.

- **STANDARD MODE** permits an unused bit in the standard AX.25 protocol to be toggled. This bit is utilized by the PAD logic, network logic, and debug logic to signify that the remote is a special user capable of accessing advanced functions.

- **VERSION SELECT** allows the local console to originate connects in either AX.25 V1 or V2. This command controls only the originating version, since the common logic automatically accommodates either version on terminating connects.

The local console CRT interfaces to the NCP via one half of an SIO, which can be strapped for multiple baud rates. The console employs a 1000-character buffer to accommodate the speed differences between incoming text and the rate of display. Because the local console need not be equipped on a node, the console logic handles cases where no hardware is present.

Table 3. Some commands and functions supported by the debug logic.

Command	Format	Description
MEMORY	M Adr1,Adr2	Do a hex dump from memory location Adr1 to Adr2.
	M Adr1	Display the contents of memory location Adr1 and then wait for new contents to be entered. Entering a period escapes this mode.
COPY	C Adr1, Adr2, Adr3	Copy the contents of memory locations Adr1 to Adr2 to the new location, beginning with Adr3.
*(OFFSET)	* Adr1	The offset (*) register is useful for input arithmetic. For example, setting the * register to 1000H (*1000) allows subsequent entering of M* + 2 to display location 1002H.
HEX ARITH	H Num1 + Num2 - Num3	Perform hex arithmetic on any entered numbers. Addition and subtraction are supported with support of the offset (*) register.
PORT	P75	Display the contents of the Z-80 I/O address 75H, then wait for a new number to be input. Entering a period escapes this mode.
INITIALIZE	I Adr1, Adr2, Num	Initialize the memory block from Adr1 to Adr2 with the number (Num) entered.
EXECUTE	E Adr1	Transfer execution control to Adr1 with the Z-80 registers initialized per the "R" command.
REGISTER	R	Display and allow change of the Z-80 register file
X DIAGNOSTIC	X Adr1, Adr2	Run read/write/verify memory diagnostics on the memory block Adr1 to Adr2.

debugger

The debug logic (see *H* in **fig. 1**) has proved to be an invaluable aid in initial software debug as well as in system integration. The debugger at any node is accessed by specially authorized remote users via requesting a connect to the node utilizing a given SSID (usually -6). This logic supports a single user and inhibits others from connecting if someone is currently active.

The debugger executes concurrently with the network logic and the PMS, as well as at the same application level (see **fig. 1**). Accessed from the common logic, it is used as an aid in debugging these and other applications in an on-line manner.

Table 3 describes some of the commands and functions which are currently supported by the debug logic.

digipeating

Unless inhibited from doing so by the database, every network node can also function as a local digipeater (see *I* in **fig. 1**) for whatever frequency is being used as an input. In Dallas, it's 145.05. Multiple digipeater addresses are supported, thus allowing for any number of ALIASES or standard AX.25 addresses. It should be noted here that the PAD addresses mentioned above and the digipeater addresses are completely separate.

Enhanced digipeating is supported by the node because the output physical channel for retransmission of a packet is a function of both the digipeater address and the physical channel on which the packet was received. Thus, it's possible to have a cross-frequency digipeater using no special tricks or logic. For example, N5EG-8 (Node 4, MURPHY) is configured to be a bidirectional cross-frequency digipeater between 2 meters and 450 MHz. This has proven to be extraordinarily useful in network software construction because it allows access to a remote node (one which is potentially many hops away) via digipeating over the 9600-bps trunk circuits. This method of access completely bypasses all other software and is, therefore, useful (in conjunction with the debugger) in troubleshooting.

conclusion

Thanks to the members of the Texas Packet Radio Society for helping to make this series of articles possible.

We'd like to hear from developers and users of other packet systems to learn what you're doing. Please address correspondence (enclose SASE) to Tom McDermott, N5EG, The Texas Packet Radio Society, P.O. Box 831566, Richardson, Texas 75083-1566.

references

1. Thomas H. Aschenbrenner, WB5PUC, and Thomas C. McDermott, N5EG, "The TEXNET Packet Switching Network: Part 1 — System Definition and Design," *ham radio*, March, 1987, page 29.
2. Thomas H. Aschenbrenner, WB5PUC, and Thomas C. McDermott, N5EG, "The TEXNET Packet Switching Network: Part 2 — Hardware Design," *ham radio*, April, 1987, page 29.
3. Bill Wade, WD5HJP, "Packet Radio Conference Bridge," *ham radio*, April, 1987, page 24.

ham radio

packet radio conference bridge

AX.25-compatible bridge links six stations for routine or emergency communications

From the beginning, packet radio has been essentially a point-to-point mode. Even with the TNC-2's multi-connect capabilities, it hasn't been possible for all participants in roundtable or net-type operations to be connected to everyone else in the net. Although makeshift arrangements have been devised, they've lacked the anti-collision and error-controlling capabilities of the AX.25 protocol.

In an effort to solve these problems, Tom Aschenbrenner, WB5PUC, has designed a truly error-controllable, AX.25-compatible conference bridge. Because it's combined with other network software components he's designed, the conference bridge is offered in two versions. The one described here fits into a replacement EPROM for a TNC-2 clone; the other is a part of the software installed in the 9600-baud network node controller board for TexNet.[1,2]

A conference bridge module in a TNC-2 clone or a TexNet node provides full-protocol, multiple-station roundtable or net-type operation between packet stations. Typical operation is accomplished by each of the stations involved in the net connecting to the bridge-equipped node by using the Secondary Station Identifier (SSID) assigned to the conference bridge function. On the test nodes in operation in the Dallas area, the SSIDs are -2 and -3 on each node. The current version of the network software supports two independent conference bridges of six participants each. It's also possible to connect to the bridge through one digipeater if necessary.

typical conference bridge operation

To connect to a conference bridge, each station connects as if it were connecting to any other packet station. A typical text sequence to a conference bridge would be:

C WB5PUC-2 <carriage return>

The operator's TNC does the connect routines.

The following will then appear on the operator's CRT:

*****Connected to WB5PUC-2 <cr lf>**
(from TNC)
Welcome to the WB5PUC Conference Bridge. A Control-U shows all stations connected to this bridge.

At this point or at any other time, the response to a Control-U command to the bridge will bring up a text string listing the calls of all connected stations. For example:

N5EG-5 WD5HJP W5YR-7 WA5MWD-3
connected

No additional commands are needed to operate the bridge.

In the normal operation mode, each operator receives a text string with a shorter header indicating the call of the originating station. For example:

WD5HJP> Transmitter power is now at 100 watts.
N5EG-5> OK Bill, try adjusting trimmer C15 now.
W5YR-7> What are you guys up to?
WD5HJP> Hi George, just adjusting the node's final amp.

When the QSO is over, those connected to the bridge simply disconnect as they would from any other packet connection, via a DISCONNECT command in the Command mode of the TNC.

operations test

A routine test of the emergency plan for the Point Beach, Wisconsin nuclear power plant in September, 1986, provided an intensive on-the-air test of an early version of the conference bridge software. Though not specifically designed for such use, the conference

Bill Wade, WD5HJP, Texas Packet Radio Society, 600 Via Sevilla, Mesquite, Texas 75150

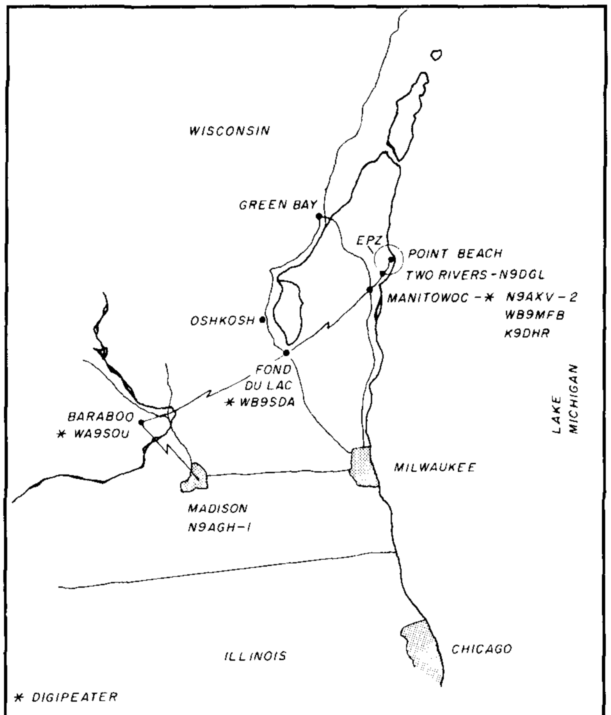


fig. 1. Conference bridge linked six stations in nuclear plant emergency exercise.

bridge served as the hub of an emergency communications network that included a link to the state capital. Overall, the bridge performed well in its original form; later modifications have been implemented to facilitate its use in emergencies.

The Nuclear Regulatory Commission (NRC) requires annual testing of every nuclear power plant's emergency plan. This test is designed to evaluate the ability of plant personnel and the utility company holding the facility's license to cope with an accident. The NRC measures their ability to assess the extent of danger to the public, their effectiveness in recovering control of the plant, and their ability to minimize damage to the surrounding environment. A succession of events pushes the plant engineering staff through a series of critical decisions; events are programmed into the scenario to simulate damage to the power plant and motivate recovery action by the staff.

For a realistic overview of the performance of the allied agencies that would be involved in an actual incident, the test scenario includes a simulated evacuation. State, county, and municipal emergency units become involved in the plan when supervisory NRC engineers and the utility's power plant engineers have recognized a possible threat to public safety. At some point in the escalation of the situation, the plant staff recommends evacuation, triggering activation of a number of government safety, information, and public assistance centers.

At that point, the power plant staff contacts the county and state emergency units and delivers its assessment of the situation. The complement of offices that become active in the evacuation phase of the emergency plan are the county Emergency Operations Center (EOC); the state Emergency Government Office (EGO); the Department of Human Resources Reception Centers that will process evacuees; the Joint Information Press Center (JIPC), the official information center for the emergency; and the off-site power plant Emergency Operations Facility (EOF), from which the NRC and utility supervisory engineers make their recommendations to the state and county emergency government.

Each of the 102 nuclear power plants in operation in the United States is surrrounded by an Emergency Protection Zone (EPZ) with a radius of 10 miles. This EPZ is the area considered under immediate hazard in case of any airborne release of radioactivity. The Point Beach power plant EPZ includes three towns — Two Rivers, Shoto, and Two Creeks — and extends into Lake Michigan.

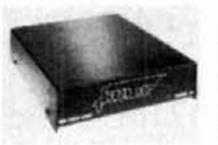
test scenario

This particular scenario began with a hypothetical earthquake tremor, causing a series of leaks in the cooling loops of the "B" power plant. Subsequent "damage" to the plant caused some injuries and contamination of plant employees and allowed the release of radioactive steam into the atmosphere. By 8 AM, after a damage assessment, the county and state government were notified that there was a "public danger"; the county emergency office notified the Manitowoc RACES EC, who activated the amateur emergency system. Stations equipped with 2-meter fm voice and packet equipment were established at each of the operations centers listed above, with the exception of the plant Emergency Operations Facility (EOF), not normally accessible to the public.

RACES participation

The Amateur community in Manitowoc County is actively involved with the county emergency government office. During the test, the five-station RACES system was centered around the Manitowoc County EOC, which has a permanent Amateur Radio station linked via microwave to a dedicated voice repeater (WB9MFB, 145.19/.79) located at the county transmitter facility. Four remotely-sited, linked voting receivers increase the effectiveness of this 60-mile diameter service area, 2-meter fm voice system. For packet operations, a permanent conference bridge-equipped packet digipeater (N9AXV-2, 145.01 MHz) utilizing an MFJ-1270 TNC-2 is installed at the transmitter site. The voice repeater and packet digipeater are converted 100-watt Motorola transceivers.

ANNOUNCING
SOUND OFF MODEL 10



Sound Off Model 10 is a self-contained message generator complete with case and power supply. The Model 10 was designed specifically for applications where repetitive high quality natural voice announcements are required, i.e. Amateur repeater and commercial radio systems. The Model 10 stores your natural voice message in nonvolatile EPROM with single message lengths of up to 32 seconds. The Model 10 provides for both single and continuous play modes. Factory setable internal timers can set timing intervals from seconds to 30 minutes or more. All control functions and timers are provided for this application. (COR sense, transmitter activity sense, PTT, ID timer.)

Models 20, 100 and 150 available for applications requiring random access multimessage capabilities in nonvolatile re-recordable message times of up to 4 minutes.

Model 10 prices start at $219.00.

Now for the first time you can enjoy the truly unique operation of a Digital Voice Announcement System, designed specifically for Amateur Radio communications. The DVK-100 represents the latest technology in digital audio processing.

Create your own natural voice contest calls, CQ's etc. Your voice is stored in digital memory, ready to be played back at the touch of a key. The Digital Voice Keyer is not a tape recorder or robotic sounding synthesizer but a true full fidelity natural voice record/playback system.

The DVK-100, is a must for the avid contester and great audio accessory for any Ham Shack.

FEATURES

- Superior natural voice quality
- Micro-processor controlled
- 32 seconds of message time
- PTT/VOX operation
- Dynamic/condensor mic input
- Selectable monitor amplifier with preset level controls
- Selectable audio compressor
- Sealed membrane keyboard
- 4 independent voice memories
- Positive/negative keyed PTT
- ESD/EMI/RFI shielding
- Selectable end of transmission tone generator

✓ 185

The sound of the future is here today. Contact your local Amateur Radio dealer or NTL for further information.

$249⁰⁰ SUGGESTED NET AMATEUR PRICE

DEALER INQUIRIES INVITED

NEL-TECH LABS, INC.
28 DEVONSHIRE LANE • LONDONDERRY, NH 03053
603-434-8234

HAM-COM 1987
&
ARRL
TEXAS STATE
CONVENTION

JUNE 5, 6 & 7 1987
ARLINGTON CONVENTION CENTER
ARLINGTON, TEXAS

10th Annual Ham-Com!

More Than 6,500 Registrants Expected!

Be There!

Exhibitor Info: (214) 521-9430
Registration Info: (214) 423-3498

Other equipment used at all of the temporary public service Amateur stations included additional 2-meter fm voice transceivers for the RACES 145.19/.79 repeater. Voice fm was used to coordinate large file transfers and to back up the packet equipment. A variety of computers (chiefly IBMs and IBM clones and Commodore 64s) and TNCs (AEA, Kantronics, MFJ, and original TAPR 1's and 2's) were used without any compatibility problems. Each of the locations was equipped with the appropriate disk drives; some were also equipped with printers.

Operators at each of the centers connected to N9AXV-2, the conference bridge/digipeater at the county transmitter site. N9AGH-1 was at the Wisconson State Emergency Government Office in Madison, the state capital. To reach the conference bridge, N9AGH-1 connected through two digipeaters, WA9SOU (in Baraboo) and WB9SDA (in Fond Du Lac), a link length of over 150 miles (see **fig. 2**). WB9MFB, at the county EOC, K9DHR, at the reception center in Manitowoc, and N9DGL, at the JIPC in Two Rivers, were all connected directly to the bridge.

Once the stations were connected, typical operation proceeded as in any other net. Net control was maintained by operators at WB9MFB, the county EOC. Traffic was passed simply by sending the text via packet from each of the sites. Since all stations on the bridge were getting identical copies of text, very little repetition was necessary.

impact of packet operation

The worth of the conference bridge and packet radio was demonstrated immediately. The test emergency was declared from the governor's office in Madison, and the message went out via the two-digipeater link to the conference bridge at the Manitowoc EOC. A parallel 75-meter phone system running from the state emergency office to the county EOC was also activated. The packet conference bridge delivered the message correctly, approximately 30 seconds ahead of the 75-meter phone link. The 75-meter system garbled the power plant emergency protection zone (EPZ) grid coordinates in the first message!

In some respects, use of the packet conference bridge became second nature. To simulate a system failure, Ron Shimek, WB9MFB, the county EC, disconnected power to the conference bridge, forcing the use of point-to-point packet communcation. When they became aware of the system failure, packet operators set up point-to-point links to re-establish communication. Although throughput was slower, point-to-point operation did provide usable information. As a backup, a standby station had been set up to monitor activity and act as recorder for the entire test.

test results

At the conclusion of the exercise, the Emergency Coordinator described the conference bridge as a key ingredient in the success of the test. In his report, he emphasized that federal observers had been impressed not only with the speed and accuracy of traffic handling, but also with the capability for time/date stamping of traffic (using the computer systems' real-time clocks) and for producing hard copy simultaneously at all sites.

Because all the stations in the emergency system were connected through the conference bridge, automatic status and warning updates were available at all sites. An NRC inspector mentioned to one of the participating Radio Amateurs that thanks to ham activity, he was never out of touch with any of the sites for the duration of the test. The Federal Emergency Management Authority (FEMA) observers indicated that they would encourage further use of packet radio and the conference bridge in future tests. WB9MFB's evaluation of the test suggested a broader role for the RACES group in both forthcoming tests and actual emergencies.

recommendations for future designs

The relatively few operation anomalies that caused some delays on the system were largely attributable to QRM originating outside the limits of the test area. The Manitowoc county transmitter site is about 3 miles from Lake Michigan; there is considerable channel activity from Milwaukee and Chicago to the south and from Michigan across the lake. All of the test activity took place on 145.01 MHz. Typically, there are periods of 10 to 20 minutes when the N9AXV digipeater squelch never closes. Selection of another frequency besides 145.01 MHz for operation would be desirable.

Connection of the conference bridge to a network system is also recommended. In this test, the state emergency government center was by necessity connected through a 145.01-MHz, two-digipeater link. Response time from that part of the system was proportionately slower, but still quite usable. Other situations without such a robust digipeater link would benefit from use of a backbone network system. The conference bridge software does allow connection to any network because of its compatability with AX.25.

WB9MFB has suggested that a system monitor be set up in advance to record all activity during an emergency communications test. Packet operation offers a significant advantage in this regard, in that all text information is easily stored to disk. In this case, the system monitor provided a good backup to the NCS; later, when the EC and RACES group needed to do evaluation of their own, the stored information proved to be useful indeed.

Other recommendations include eliminating BBS activity on frequency during tests or actual emergencies to help avoid channel congestion. Operators from outside the test area occasionally interrupted RACES activity with inquiries about the test and the conference bridge.

implications of conference bridge operation

Amateurs who've handled traffic in hurricanes, tornados, or other disasters know that telephone systems are the first communications systems to become overloaded or destroyed. The Point Beach scenario demonstrated that utility companies and state agencies are still blindly tied to this relatively fragile communications resource. The ability of packet radio systems to handle high volumes of traffic quickly and accurately under difficult circumstances is being demonstrated regularly; as part of these systems, a conference bridge can provide reliable communications to the people that need it most.

references

1. Thomas H. Aschenbrenner, WB5PUC, and Thomas C. McDermott, N5SEG, "The TEXNET Packet-Switching Network — Part 1: System Definition and Design," *ham radio*, March 1987, page 29.

2. Thomas H. Aschenbrenner, WB5PUC, and Thomas C. McDermott, N5SEG, "The TEXNET Packet-Switching Network — Part 2: Hardware Design," *ham radio*, April, 1987 page 29.

ham radio

GLB PACKET RADIO GOES PORTABLE

THE FIRST CONTROLLER DESIGNED FOR PORTABLE AND SOLAR-POWERED STATIONS

- LOW 25 mA Current Drain.
- Miniature size - Lightweight.
- Rugged metal, shielded case.
- Lithium Battery backup for RAM.
- Onboard Watchdog for reliability.
- Standard DB25 Connectors.
- "Connected" Status output line.
- Remote Commands in Unattended Mode with Hardware Lockout.
- Retains all other PK-1 features.
- Extra I/O lines for special applications.

NEW SOFTWARE FEATURE:
INTELLIGENT "BUDLIST" - Provides selective callsign filtering for Digipeating, Monitoring and Connecting.

Model PK1-L
Wired/Tested
List price — $209.95
Amateur net — $179.95

Power requirement: 9 to 15 Volts DC @ 25 mA typical
Dimensions: 4.6 X 5.9 X 1.0 inches **Total Weight:** 12 ozs.

See us at
Dayton
Booth 318

Please specify Call Sign, SSID Number, and Node Number when ordering.

Contact GLB for additional info and available options.
We offer a complete line of transmitters and receivers, strips, preselector preamps, CWID'ers & synthesizers for amateur & commercial use.
Request our FREE catalog. MC & Visa welcome.

✓ 186

GLB ELECTRONICS, INC.
151 Commerce Pkwy., Buffalo, NY 14224 716-675-6740 9 to 4