PACKET STATUS REGISTER

Tucson Amateur Packet Radio Corporation

President's Report

Although summer is ending, TAPR is getting geared up for the remaining half of the year and starting to plan 1994. Let me attempt to summarize what has been happening since the annual meeting.

The TAPR/AMSAT DSP project is moving forward. We should have a full article on the new design in the next PSR. No commitment on dates, but we have determined that we will be doing a limited first run for beta-test units before we do the final kit production (see article inside).

The TrakBox kits have finally been finished. Due to summer vacation schedules and getting in the last few hard-to-find parts, we were about a month off from our initial guesstimate on kit sales.

Paul Newland, AD7I, has approached the board with a proposal for a new project called PCON. PCON (A Simple Printer Controller for Parallel Printers and Serial Mechanical Teleprinters) has been accepted by the board and Paul is continuing his development. PCON, in its most basic form, is a printer controller that serves to convert serial asynchronous ASCII data to a format suitable for driving a parallel printer (or ASCII/BAUDOT serial asynchronous teleprinter). A typical use for PCON would be to interface a parallel printer to a packet radio TNC. Such a system provides the functionality of a "rip-and-read" message printing system where the person receiving the message doesn't need to know anything about radio or computers. Numerous emergency applications are possible. Paul will be providing a full write-up for the next PSR. If you have a project that you want TAPR to examine for possible release as a kit or for technical feedback, refer to the article on submitting projects.

The TAPR e-mail server has been set up and is now operational. Lou Nigro, KW7H, is administrating the system. Lou is also the TAPR software Librarian. The board has moved off CompuServe and now uses the mail server for all of its communications. There are a few more things to be worked out, but over-all the change has been a good one. We will be setting up a mail distribution system in the near future in order to disseminate information between PSR issues.

Although sales from the office have been slow this summer, financially, TAPR is still showing a net gain for the year. This is a direct result of everyone in the organization paying special attention to expenses. We hope to continue this trend and keep things in the black for the entire year. Don't forget the office is open Tuesday through Friday from 10am till 3pm MST. Heather always likes to hear from you, so give the office a call if you need anything. In order to help keep membership dues up-to-date, TAPR will begin to mail out renewal letters from time to time. We all understand it is sometimes hard to see that your membership has expired from receiving the PSR, since most folks don't prioritize reading the mailing label (grin). This mailing should help to make sure no one misses out on any TAPR information or the PSR.

In the area of TAPR meetings, a date for the annual meeting in Tucson has been set, so please mark your calendars now and keep your eyes open for discount airline

tickets to Tucson. The meeting will be held the first weekend in March (5th & 6th). There will be a mailing to all members in the Fall with in-depth information. There will be a proceedings printed for the meeting, so start thinking ahead about how to write up your current project or research. TAPR will also be attending some additional ham conventions throughout the year along with our traditional Dayton visit. TAPR will be at the August convention of HamShows held at Valley Forge, PA.

To finish this issue off, I want to touch on membership. Membership is the life-blood of any organization. TAPR's future depends on new members. New project ideas, software developers, network designers, policy writers, writers of articles for the PSR, and the future board comes from the membership. If you have any ideas on how TAPR can increase membership in your area, let us know. If you have someone who you think would like to join TAPR or needs information on the organization, please contact the office and we will get a *PSR* and some membership information in the mail.

TAPR Project Proposal Format

One of the primary purposes of TAPR is to provide input and support for research, development, and standards for new applications and technology. In order to effectively provide an avenue for new project support we have developed an outline for proposal submissions. This is based on the earlier submission format from a few years ago, but is much more simplified. The main purpose of the document is to provide enough information on the project in question, so that the board can make an informed decision. Hopefully the form is pretty much self-explanatory. The whole submission shouldn't be more than about 5 pages total. Proposal submissions should be sent to the TAPR office or e-mailed to Greg Jones (wd5ivd@tapr.org). Notification of receipt will be sent as soon as distributed to the board. The board will then discuss the proposal in a timely fashion and provide feedback. Again, we hope this will facilitate project concepts and ideas within TAPR. If you have any suggestions or comments on this format or process, please let us know.

1. Cover Page

   a. Name, Address, Phone, Fax, E-mail of project participants. (please indicate project leader)

   b. Title of Project or Proposal

   c. Time Period of Project

   d. Total Amount Requested

2. Technical Abstract of Project (short overview — less than a page)

3. Research/Project Objectives

4. Research/Project Impact (who this is intended for and what is the potential application/service/function)

5. Research/Project Personnel (briefly describe the participants and what they will be doing)

6. Technology Transfer (what is the potential for something TAPR would develop, kit, propose, or publish in the future)

7. Budget Justification (spreadsheet)

8. Attach anything else you think necessary for the proposal.

Notes from the TAPR Office

We are now on the Internet System from the office. This is due to the hard work of Lou Nigro, KW7H, and Bob Nielsen, W6SWE. I'm sure the thanks should go to additional volunteers, but these are the two that interface with me at the office, and I have witnessed, and am benefitting from their time and energies. They have made it so painless that I am thoroughly enjoying it! Thank you.

Those of you that have had problems with your Deviation Meter at the testing point, We have found that the small PAL chip labeled DEVMTR, in some cases got programed with a corrupted master, causing these dificulties. Please return this chip and we will quickly send you a replacement.

For those of you who send us a FAX from overseas, please help me with your city and country codes. There have been a few of you who have thought that I did not answer your FAX on purpose. To the contrary, I have stood at the FAX machine and tried every combination of numbers that my telephone directory will offer me on your country, and it has refused to produce results. Your help with this will make me more efficient in getting you the material that you are requesting. Also, please do not give me any "extra" numbers that only pertain to communicating within your own country. I have no way to know which they are so they only add to the confusion!

I heard a really nifty story concerning morse code a while back that I thought you might enjoy reading about. There is a married couple who both are hams, and both proficient at morse code. The wife found herself in a hospital setting, unable to communicate with the hospital staff, due to her severe condition, and all the apparatus that she was attached to. The husband came to her bedside and found that as he held her hand, she was sending him code-by-touch. The staff, never before having been able to know what a patient in this condition desired, was fascinated as they heard from the husband what she was saying. Isn't our hobby NEAT!

Our trip Down-Under went well, very enjoyably actually! And we hear from our son at West Point that things, although ten times harder than he expected them to be, is settling in, and taking on the challenge.

73
for TAPR,
Heather, N7DZU

TAPR/AMSAT DSP Project Update

The TAPR/AMSAT DSP project is well under way. Initial prototypes of the new design will be under test the first of September. After the alpha-testing is completed, an initial beta-test run (approximately 50 kits) will be done in order to test the final design and allow software developers to test the units. The TAPR/AMSAT DSP unit is based on the TMS320C25, and supports a high-speed serial interface. A full write-up on the design will be printed in the next *PSR*. The DSP project is now accepting information from folks interested in participating in the beta-testing phase. The purpose of the beta-test is threefold: first and most important, create a base of developers that can test initial software and begin to develop future software for the unit; second, provide feedback on the kit and check various operational parameters, and third, allow software developers to add support to their existing applications. We are looking for software developers who need to adapt their software with the unit or individuals who have experience with programming assembler routines and have some knowledge of DSP to test the initial software and begin to develop new software. If you think you are interested, please write the TAPR office stating your qualifications and the reasons you would like to participate in the beta-test. Be sure to include information on e-mail, address, and various phone/fax numbers. Final decisions on beta-tester selection will be made later this year by the development team.

The TAPR Board of Director members "attend" a meeting, which is continuously in session, on a private bulletin board system. The Board encourages input from all interested members. If you have an issue you want addressed, or an idea for a project you would like TAPR to sponsor, contact any Board member, or drop a note to the TAPR office.

TAPR is now accessable through the Internet. You may send e-mail messages (no long files, please) to the TAPR office at

tapr@tapr.org

and to any of the directors at

callsign@tapr.org

substituting their call for "callsign." Also, submittals for *Packet Status Register* may be sent to

psr@tapr.org

in addition to the CompuServe address on the front cover.

The TrakBox Project

By Lyle Johnson, WA7GXD

Have you ever tried to operate an OSCAR or RS satellite in low earth orbit (this is every OSCAR except AO-10 and AO-13)? If so, you have probably wished for three or maybe even four arms.

You have to point your directional antennas at the satellites which are rapidly moving across the sky. This means updating both azimuth and elevation every minute or so. You have to constantly retune your radio so your uplink and downlink stay where they belong. And you have to fiddle with your microphone or keyer, and maybe switch antenna circularity.

You can do it manually (I have!), but it isn't too much fun.

Now, if you have a PC or clone, and if it has available slots, you can always buy a Kansas City Tracker and let it handle the antennas, and possibly the radios as well. If you have no slots, or aren't using a PC, your choices are more limited. You have to use the armstrong method.

Well, Bruce, SM0TER, decided this was no fun and came up with a program that runs on a MicroMint Basic 8052 controller. It is based on W3IWI's orbit tracking programs made popular in the early '80s.

This was a step forward — it didn't require a PC — but it was still expensive and a bit slow.

Sueo, JA6FTL and some others in JAMSAT, started corresponding with Bruce via UO-14. They collaborated and came up with a PC board that was much cheaper than the MicroMint controller and included everything necessary for computing satellite orbits and pointing antennas automatically. They wrote software for the board in 'C' and sped it up considerably.

Jack, WA4EJR, got involved with Bruce and Sueo and became a Beta tester of the new tracker. As time went on, more and more folks became interested in this device.

To shorten the story, new PC boards were made in December 1991 and a number of us built them and helped

issue bug reports, write documentation and so forth. The efforts by the JAs were underwritten by JAMSAT, and TAPR was asked to provide a distribution channel in the U.S. for anyone interested in this device.

The TAPR Board decided this project made sense. TAPR's experience in bringing useful kits to Packeteers and Satellite types was a good match, so TAPR is now bringing this project to you!

The unit is a sort-of super MetCon, with lots of digital I/O as well as an 8-channel 10-bit A/D converter. Configured as a satellite tracker, it can be as simple as the board, a source of 12 volts DC to run it, and a connection to your rotator control box(es) and serial port of your computer and let it run.

Alternatively, you can add a couple switches to it, a potentiometer and a 2-line 16-character LCD display and have a totally self-contained unit that needs no computer at all! (Well, you need a computer or terminal to initially set the real-time clock and load the satellite elements, but after that it is standalone!)

The software can control Kenwood, ICOM, and Yeasu satellite rigs for doppler correction.

TrakBox again available from TAPR

Once again TAPR is able to offer the TrakBox in kit form. We were able to obtain a limited quantity of the PC boards from JAMSAT and have gathered the necessary parts to make complete kits, less case and external connectors.

The TrakBox is a self-contained, standalone accessory for use with satellite stations. You simply select the satellite you wish to track from a front panel control, and the TrakBox will steer your antennas in both azimuth and elevation to track the selected satellite when it is near or above your horizon.

In addition, if you have a radio with a computer interface, the TrakBox will tune your transmitter and receiver and correct them for doppler shift.

TrakBox includes a serial port to
a

llow you to set the real-time clock (which is battery backed), your station location (latitude, longitude, and elevation) and the keplerian elements for the satellites you are interested in (up to 40 satellites may be loaded, and the elements may be uploaded in ASCII in either the AMSAT or NASA formats). Once set up, TrakBox no longer needs to be connected to any other computer to operate.

TrakBox includes a two-line LCD display showing the azimuth and elevation of a satellite, along with the satellite ID and GMT. The display is updated every second.

TrakBox is set up to directly interface to Kenpro, Emmoto, and Yaesu dual-axis rotators which include a computer interface. It may also be interfaced to other rotators as long as simple relays (mechanical or solid-state) are added to the rotator control box. It has been tested with the KR500 elevation rotator and the CDE/Hy-Gain series of azimuth rotators (HAM-M, HAM-II, HAM-IV, TR-44 and TailTwister styles).

The radio tuning mode has been tested with Yaesu FT-736, Kenwood TS790, and ICOM 970. It is designed to handle any ICOM radio with the CI-V interface (IC275, IC475, IC1275) as well as the Kenwood TS-711 and TS-811 models. ICOM radios with the older CI-IV interface can be used with an accessory convertor available from ICOM.

An optional tuning mode that utilizes microphone up/down buttons is also incorporated. This requires inputs from a modem (PSK or 9600 bps FSK) and is only for digital operations. This has been tested with Kenwood and Yaesu radios.

The price for TrakBox kits is $195.00, which includes a $25.00 donation to JAMSAT for support of the Phase 3D Amateur Satellite program.

The quantity available is limited and they will be sold on a first-come, first-served basis.

[If you're in the United Kingdom, AMSAT-UK sponsors a Trackbox Users Group; contact Fred Southwell, G6ZRU in Brighton for more information.]

An Automatic Doppler Correction Circuit for FSK/FM Satellites

by Mike Dorsett, G6GEJ

[Reprinted from the June/July 1993 issue of OSCAR News, published by AMSAT-UK.]

The intention of this circuit is to provide a simple and reliable method of correcting for the doppler shift (and for any frequency inaccuracy) while you look after the azimuth and/or elevation.

The circuit will work with any receiver with a logic up/down frequency control input, the original was designed for use with a Yaesu FT736R. Access to the discriminator output (DC coupled) is required. The nominal discriminator voltage for the 736 is about 4 volts. If your discriminator voltage is different from this, then the reference voltage for the two comparators will need to be altered.

How the Circuit Works

The input from the discriminator is filtered by R1/C1 to remove high frequency noise and modulation from the DC voltage that the discriminator generates. This voltage is compared to a reference voltage by each of two comparators. The reference voltage for each comparator is slightly different to prevent the circuit from hunting around on a blank channel, or on an accurately tuned signal. R2 provides a means of setting this dead band, the higher the voltage across R2, the larger the frequency error has to be before corrective action will be taken. The signals from the two comparators are gated with a slow pulse to prevent the tuning from going into scan mode. The slow pulse is generated by U4C, R9, and C4. R7/D3 and R8/D4 prevent the output from the comparators from exceeding the logic levels of U4. The frequency of this pulse determines the maximum rate at which the circuit will tune. If this is set too fast, it won't tune at all because the de-bounce circuit will eliminate the pulses!

LEDs D5 and D2 are included to show when the circuit operates.

Construction

Any method of construction can be used. The prototype was built on a plug-in board, then, a printed circuit board was developed for the final unit. Veroboard would be adequate. The unit was housed in a small diecast box. A note on components: with the exception of R2, almost any general purpose can be used. R2 MUST be a multi-turn pot, otherwise you will not be able to set it with any degree of accuracy.

Setting Up

Apply 12-15 volts to the circuit, D2 should flash. Apply 12 volts to P1, after a short period D5 will start to flash and D2 will stop. If this happens, all is well, and P1 should be connected to the discriminator output and the radio tuned to a blank channel. With R2 set to minimum, adjust R3 until both

Doppler Correction Circuit
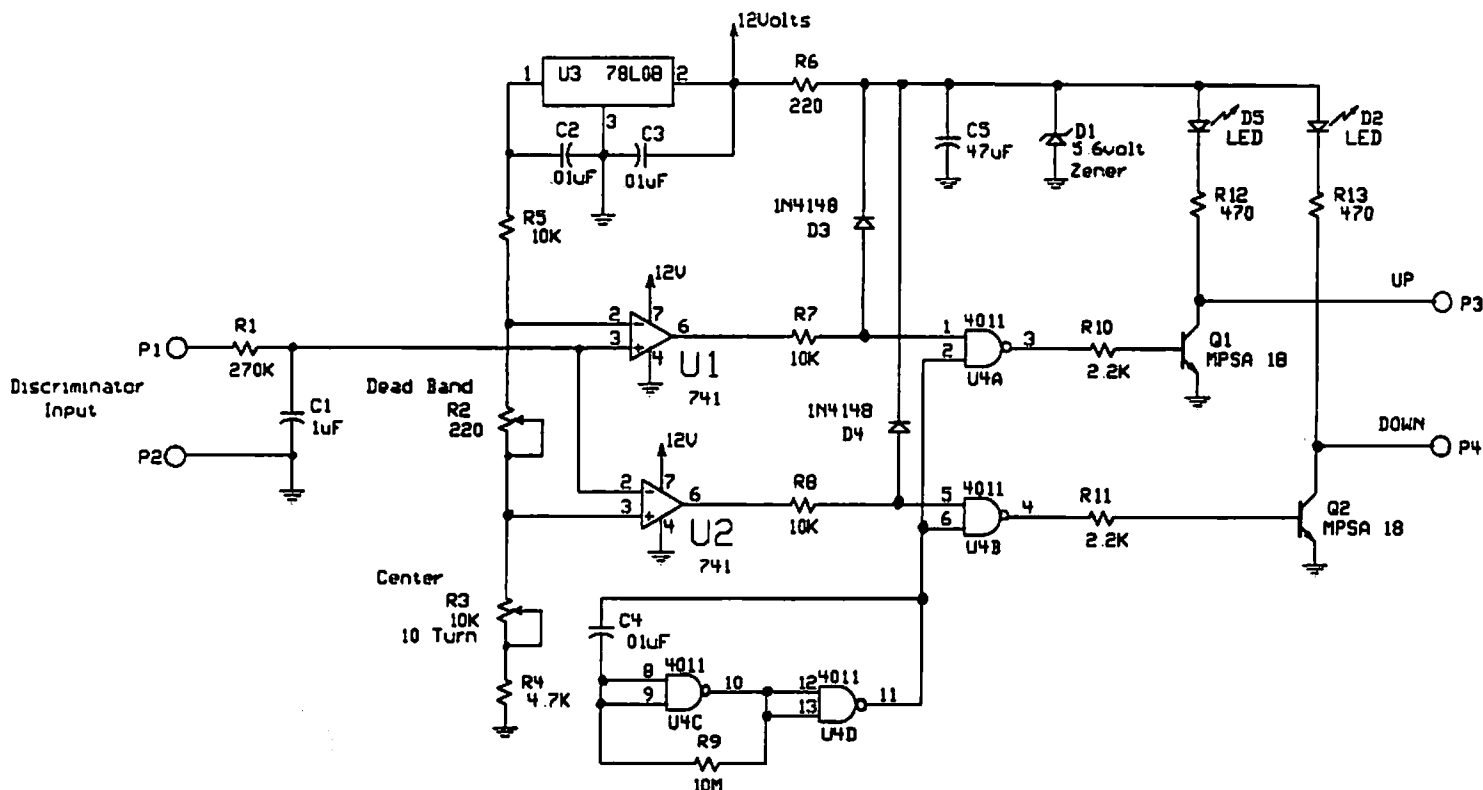
LEDs flash intermittently. Then set R2 for a voltage of between 35 to 50 mV across it. Both the LEDs should now stay out, or only flash very occasionally. Tune the receiver to a stable carrier with the discriminator meter centered, and check that the two LEDs are still out. A small adjustment or R3 may be necessary. Connect the Up and Down outputs to the receiver. If you tune off frequency slightly, one LED should flash and the receiver will return to the center of the signal; if it zooms off to a blank channel, reverse the Up and Down connections and try again.

The capture range of the circuit will depend on the bandwidth of the IF, and to a certain extent, on the discriminator characteristics. With the FT736R, the circuit will pull in a signal with a +/- 11.5 KHz offset. During tests with UO-22, the circuit had the downlink tuned in with only a few dB of signal-to-noise to work with, and then kept the signal perfectly tuned throughout the pass.

[A limited supply of printed circuit boards with this circuit are available from AMSAT-UK on a first-come first-served basis.]

RealTrak

by Michael Owen, W9IP

The recent announcement by TAPR of availability of more JA6FTL Trak-Box kits is welcome news.

RealTrak (by me) fully supports the TrakBox's "host mode." In cases where you want to track satellites that are not in the TrakBox's memory or if your want to track the Moon/Sun/noise sources/planets, RealTrak will drive the TrakBox directly. This feature has been available in RealTrak for 6 months and has been thoroughly tested by JA6FTL. RealTrak also includes a built-in dumb terminal for communication with the TrakBox's monitor program, uploading of Keplerian elements, etc.

Northern Lights Software
Star Route, Box 60
Canton, NY 13617
(315) 379-0161 (6-9pm)
FAX (315) 379-5804

ITAMSAT Project Status

by Paolo Pitacco, IW3QBN
AMSAT-I President

On Saturday, May 8, 1993, all people involved in the first Italian MicroSAT project met in Milan for the integration of all the satellite modules. The stacking was performed very well, and all modules are ready for the first vibration test. All systems are similar to the past MicroSATs except some changes were made due to upgrade or

necessity. We have introduced an FM modulator with G3RUH encoding into the transmitter module, and two demodulators (each of two chips only!) of the same style into the receiver module, for 9600 bps packet. We have changed entirely the CPU module, designed a 4 MByte static RAM, modified the transmitter module, and remade the AART.

The CPU is now composed of four 2-layer boards which contain: I/O, CPU and DMA, EDAC (256KB), and RAM-DISK (4MB). All boards fit into the original CPU frame. The transmitter module is only BPSK (twice), no RC, with one unit (TXB) switchable to FM at 9600 bps. The AART is entirely new, but has the same dimensions and electrical interface as the original; this job was caused by the fact that AARTs aren't available in SMD cases!

In the ITAMSAT Project, I am responsible especially for the transmitter module, but I have also designed the CPU boards, the AART board, and the transmitter power supply and control board.

We have prepared an experiment for this flight; there are two purposes for this experiment: to measure the attitude of the spacecraft using 10 sensors on two heads, and to measure the energy emitted from the Sun using 8 sensors.

The launch is scheduled for September 1st, from Kourou, on an Ariane 4 launcher, and the launch campaign is scheduled to start on August 1st.
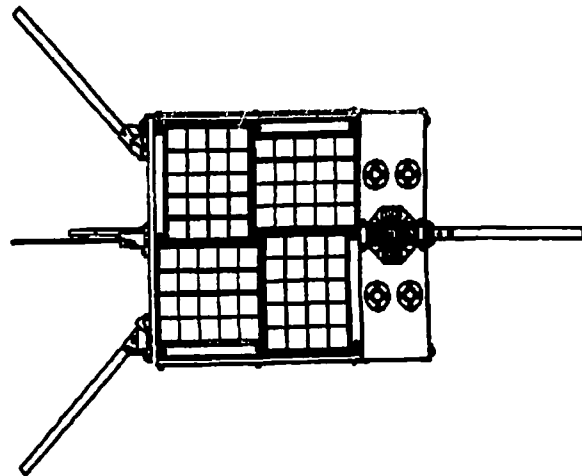
All G3RUH Modems Are Not the Same

Robert E. Dubke, K0SIR

[Reprinted from the May 1993 issue of QEX, published by the American Radio Relay League.]

In building a high speed backbone for packet, one of the nodes was to use a Kantronics Data Engine with a DE19K2/9K6 19200/9600-baud (G3RUH compatible) modem and an ICOM 38A, a 222 MHz band transceiver. This seemed like an easy thing to implement because the 38A had already been wrung out using an MFJ compatible G3RUH modem. Wrong! Upon investigation, it was found that the 38A's VCO, used for both transmit and receive modes, was being continuously modulated by the Kantronics G3RUN modem in receive mode. This rendered the receiver useless. Of three brands reviewed (Kantronics, PacComm, and MFJ), only the MFJ had the switch circuitry necessary to prevent this receive-mode condition.

The ICOM 28A/H and 38A/H models have a common design in this area and contain an audio switch for the mike. Unfortunately, it is located before a low-pass audio filter that must be circumvented for 9600-baud use. The modem's transmit audio had to be inserted at the junction of R45 (15 K ohm) and R15 (2.7 K ohm), which was AFTER the switch and filter. The receive audio was picked off at pin 9 of IC1 (MC3357P).

The solution selected was to add a simple audio switch to the Kantronics DE19K2/9K6 modem that consists primarily of a CMOS switch. See the figure. The labels on the connections are Kantronics designations. The circuit-board trace creating the audio path between U1A and C10 must be cut. In the new circuit, the 4066 CMOS switch adds a switch in this audio path. The 2N2222 transistor provides signal-level conversion for the PTT, which enables and disables the CMOS switch. The CMOS switch, a 14-pin DIP IC, may be mounted in "dead bug" fashion, or by straddling a piece of foam insulation material. The 2N2222 was soldered directly to the PC board components. There is very little room for a separate board for mounting components.

This modification should work for the Kantronics DE9600 (G3RUH compatible) modem and the PacComm NB96 series of G3RUH modems as well. Now if only the ice would stay off the 222 MHz beams, then the backbone would be perking along FB.

1994 Annual TAPR Meeting

The TAPR general meeting for 1994 will be held Saturday and Sunday, March 5-6 at Tucson, AZ. The site is the same as last year: the Best Western Inn at the Airport. If you have never attended one of these meetings, you are missing a really interesting and exciting experience. Most TAPR "business" is conducted by the Board of Directors, so these meetings are devoted almost entirely to technical reports from individuals working on various projects related to packet radio. If you are a newcomer to packet radio, don't be frightened away by fears of technical complexity. We are all hams who learn as we go. If you are working on a project, we hope you will plan to attend and tell the rest of us about it. We will provide opportunities for brief progress reports in addition to more extensive presentations. In addition to the "contributed paper" sessions we have had in the past, we plan to have a symposium of invited speakers who will present several forward-looking topics on digital communication. If you know of someone (including yourself!!) who you would like us to consider for that symposium please let us know. Last year's meeting featured a valuable and in-depth workshop on Digital Signal Processing. We plan to present another workshop in 1994, and solicit your suggestions as to the topic. Send your suggestions for the symposium and/or workshop to Keith Justice, KF7TP, c/o TAPR in Tucson, or to KF7TP@tapr.org on internet. See you at the meeting!!
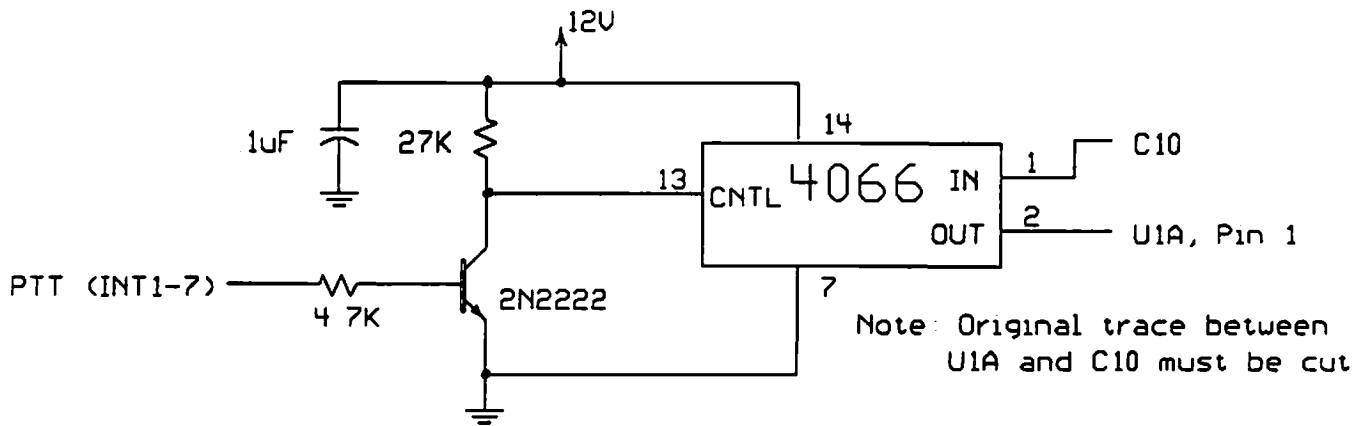
1993 ARRL Conference on Digital Communications

Who: All interested in Amateur Radio Digital Communications and Networking topics

What: A weekend of fun, information, sharing of ideas, etc.

When: September 10-11, 1993 (Friday evening and Saturday)

Where: Holiday Inn Airport in Tampa, Florida at 4500 W. Cypress

Why: Why not?!?!

Hosts: The Tampa Local Area Network (TPALAN)

Cost: Conference registration is $35.00 prior to August 21, $40.00 after August 21st. Registration covers conference proceedings, catered lunch on Saturday, and assorted items. All participants must take care of all other expenses.

Address: Send registrations to TPALAN, 6403 N. Paddock Ave., Tampa, FL 33614

Info: All who pre-register by August 21st will be sent an information packet with up-to-date information on schedule, etc. Also information on other area attractions will be sent.

Queries: Queries for information can be sent to "brianlantz@delphi.com" or to the above address.

Spouse: Not forgotten! The Hotel is a couple of blocks from a major mall, and many other stores where they can spend the family's hard earned money. There are many "distractions" that can keep them busy for the day.

Distractions: It's Florida, of course there are many! Golf course close, Busch Gardens within 20 minutes, historic sights, beaches, all within easy traveling distance. Of course, there's Disney World, Sea World, Universal Studios, and much more in nearby Orlando, only a little more than an hour away.

All Conference activities will be held in the Holiday Inn.

Shuttle service is available from the airport. Talk-in will be available on the 147.105+ Mhz repeater on both Friday and Saturday. The Hotel is right off Interstate I-275.

Hassle-free hotel rooms can be secured by calling 813/879-4800. Mention the Conference for a special rate is $50.00 a night (single or double), and you can also come up to 3 days early and stay up to 3 days after at the same rate. Do NOT use any other Holiday Inn phone numbers other than this one. This rate is guaranteed only through August 8th, so don't wait too long! (You can probably get the rate after the 8th, but you will be taking your chances.)

Tentative Schedule

Friday 10th:
4:00 p.m. Registration in hospitality suite

7:00 p.m. Birds of a Feather - roundtable discussions on various aspects of networking; i.e. TheNet, ROSE, TCP/IP, etc.

Saturday 11th:
7:00 a.m. Registration open

8:00 a.m. Conference begins

12:00 p.m. Lunch

5:00 p.m. Conference concludes

7:00 p.m. Discussions on various "Standards"; i.e. BBS forwarding, Network routing, message addressing, etc.

Sunday 12th
Enjoy Tampa, or be a party-pooper and start back home!

Valley Forge HAMSHOWS 1993 Convention

TAPR will be showing at the first of the 1993 HAMSHOWS, which will occur at the Valley Forge Convention Center, King of Prussia, PA, August 21st and 22nd, 1993. TAPR has traditionally attended Dayton each year, but the board felt that this limited access to TAPR to folks who could attend the annual meeting in Tucson, or those who could attend Dayton. This will be the first of several conventions that TAPR will start to attend in order to raise TAPR's visibility. We will be publishing a list of other planned conventions in the near future.

If you are in the Valley Forge area, please plan to attend the convention and stop by and say hello. TAPR will be presenting a session on Beginning Packet Radio and an Introduction to Digital Signal Processing, along with having a full range of TAPR kits, software, and publications for sale. We will be planning an informal dinner Saturday night, so make sure you get the time and place at our booth. If you would like to help work the booth for the show, please contact Heather at the TAPR office. For more information on the convention itself, you can call (913) 422-4646 or write to Ham-Shows, P.O. Box 3865, Shawnee, KS, 66203.
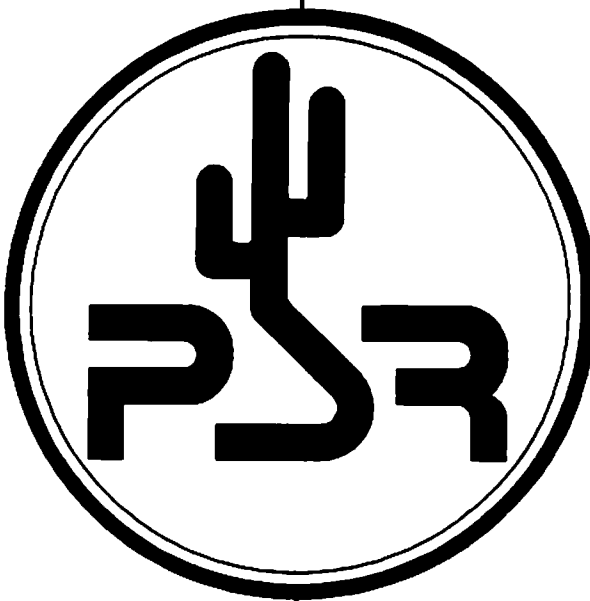
TNC-2 and the TAPR 9600 bps Modem

by Don Werts, N7NKJ

Purpose

To document the minimum parts required to use the TAPR TNC-2 and a TAPR 9600 bps modem in a full-duplex packet repeater. The TNC is used with the standard firmware (vers. 1.1.8) to supply the repeater ID function.

Modifications (Cuts and Jumpers)

(I don't know the PWB revision level, some boards may not need jumpers 1 and 2.)

1. Add a wire from the ground trace near U9 (74HC14) to its pin 7.

2. Add a wire from the ground trace near Q3 (7805) voltage regulator to its middle terminal.

3. Connect pin 11 of U3 to a close-by ground trace. We found that we did not deed the -5 volt supply to run the RS-232 port at 4800 bps. You may need to verify this for your application.

4. Cut jumpers on J4, the modem disconnect header, at terminals 11 and 12.

5. Install a jumper or shorting block on JMP12 for the A13 address line, if you're using the TAPR firmware in a 27C256 EPROM.

Parts List for the TNC-2

TAPR TNC-2 bare printed circuit board

10 pf C51
60 pF Trimmer C47
100 pF C61
220 pF C60
.01 uF C2, 19, 24, 59
.1 uF C1, 6,13, 14, 15, 17, 21, 23, 25-30, 37, 40, 41, 48, 49, 50, 61, 62
1 uF C20, 58
10 uF C8, 18, 22, 31
100 uF C16
1000 uF C12

10 uH L1, 2

LEDs CR17, 18, 19, 20, 21

1N4746 CR1
1N754 CR8
1N4752 CR12
1N4001 CR7, 22
1N4148 CR9, 10, 11, 14, 16

100 ohm R16-19
470 ohm R34, 35, 53, 54, 97
1K R11, 33, 36, 46, 47, 48
4.7K R32, 98
10K R9, 10, 12, 22, 23, 26, 28, 29, 30, 49, 50, 51, 67, 68, 69, 71, 75, 96
27K R25
47K R13
100K R20, 21, 24, 27, 31, 37, 43, 64, 72
1 Megohm R83

2N3904 Q1, 4, 6, 7, 13, 15
2N3906 Q5
VN10KM Q10
7805 Regulator Q3
Crystal Y1

U1 CD4040
U3 LM324
U4 74HCT393
U5 74HCT374
U6 2764 EPROM, State Machine
U7,9,10,14 74HC14
U11 74HC107
U12 74HCT139
U13 CD4066
U21 Z80SIO
U22 Z80A CPU
U23 27C256, TAPR 1.1.8 Firmware
U24,25 6264LP or 43256L

Miscellaneous hardware consisting of IC sockets, headers, and jumpers have been omitted from this parts list, and it is left up to the builder's preference. Please note: to install the 9600 modem onto the TNC-2 does require the 20 or 26 pin mating connector, or other suitable means of interconnection (i.e., a short ribbon cable).

Note: Q13 and its associated parts were installed to facilitate initial checkout. If you do not install this capability, R68 is still required.

Application

The TNC-2 has been running with the TAPR 9600 bps modem for two months flawlessly. This is an excellent combination of products for this application, certainly very cost effective. This configuration does not support any 1200 bps operation! The modem was built with the optional internal clock and bit-regenerative circuits. This provides a full-duplex regenerative repeater for 9600 bps packet.

The radio utilized in the repeater is a standard GE MASTR II without any modifications to the transmitter or receiver, with the following exceptions: The output from the 9600 modem is applied directly to the transmitter varactor diode, and the receiver's audio is tapped off of the discriminator. Both connections are DC coupled. The MASTR II's transmitter key-up time has been modified such that it reaches full power in less than 7 milliseconds. This last item was not a requirement, just a successful experiment! This particular MASTR II transmitter is capable of 60 watts output, which is adjusted down to 30 watts.

The repeater is installed in Western Washington at a site on Baldi Mountain (4200 ft. elevation) northeast of Enumclaw. It has been "worked" from as far away as Vancouver, BC, to South of Olympia, and into Eastern Washington. For those of you in the area, feel free to use the repeater; it transmits on 146.98 and receives on 146.38 MHz. The call is N7NKJ-8. To the best of our knowledge, this is the first full-duplex repeater with bit regeneration at 9600 bps in the Pacific Northwest.

Future Projects

As this repeater is at an elevation of 4200 ft. in the Cascade mountain range, it is usually inaccessible from December to early April; we're considering using METCON-1 to access it via the 9600 modem, or through a control link. We want to monitor the transmitter power out, the reflected power, the AC power, the backup battery status, and the outside temperature and wind speed. METCON-1 looks like a perfect fit for us.

Congratulations to all the people at TAPR for making these fine products available, and consequently making this project successful and AFFORDABLE!

Forward Error Correction

by Harold E. Price, NK6K
   Phil Karn, KA9Q

[Reprinted from the June 1993 QEX, published by the American Radio Relay League.]

There are several ways of attempting to send perfect data over imperfect RF links. One is ARQ, Automatic Repeat Request. As is done in AX.25, enough information is added to the data for the receiving station to determine whether the data was received correctly. If it was not, the computer automatically requests a retransmission. This is more efficient than the use of error-correcting codes when a data error is not likely to occur. It is required if the data must be received perfectly.

Another method is FEC, Forward Error Correction. In this scheme, you assume that there will be errors, and/or the other side of the link can't send an acknowledgment. FEC sends enough additional information along with the data to allow the receiving station to detect errors and correct them. This does not guarantee perfect data reception; the original data cannot be recovered if too much information is lost. Perfect data communication requires ARQ along with FEC.

Although an FEC mode has always been available in AMTOR, FEC hasn't been used with packet radio up to now for several reasons. First, good FEC is computationally intensive and is usually implemented in hardware. General purpose CPUs haven't become fast enough to handle high data rates until recently. Second, pure FEC adds a fixed amount of overhead. Even if the redundant information is not needed, it is always sent. On most VHF/UHF links, overhead caused by noise-created error retries is less than the fixed overhead of common FEC codes. Most metropolitan area retries are caused by collisions.

Finally, the ubiquitous TNC, with its HDLC chip, hard CRC checking, and AX.25 protocol, does not lend itself to easy FEC upgrades. We'll address the first of these problems in this month's column.

Compute power is now more readily available. FEC is put to good use in the Amateur world for links with a very narrow margin, as CLOVER does on HF. It can also be used when the link margin would be good if not for the presence of an interfering signal, as is the case in Southern California with military radar on 70 cm. You can move away, as I did (it was one of the reasons, anyway), or you can try to do something about it, as Phil Karn, KA9Q, explains below. Phil recently moved to Southern California. He spoke on FEC at this year's TAPR meeting, and has provided the summary of that talk that appears below.

KA9Q on FEC

Forward Error Correction (FEC) has been around for a long time, but only recently has it become practical for radio Amateurs to experiment with it using low cost computing equipment.

Much of the research into FEC came from NASA's need to make the most of low-power, very-long-distance communication links from planetary probes. The specific technique I will describe later was used for the Pioneer 10 and 11 probes that were launched in the middle 1970s, becoming the first man-made objects to escape the solar system. Thanks in part to FEC, these probes are still tracked by NASA's deep-space network.

Related (though different) schemes were used on later deep-space missions, such as Voyager 1 and 2.

All forward-error-correction schemes apply the same basic principle from Shannon's communication theory. According to Shannon, the capacity of a noisy, band-limited channel depends on both its signal-to-noise ratio and its bandwidth:

$$C = B \log_2(1 + S/N)$$

where

C = channel capacity, bits/sec

B = channel bandwidth, Hz

S = received signal power, watts

N = equivalent received noise power, watts

$\log_2$ = base 2 logarithm

Sometimes the signal-to-noise ratio, S/N, is replaced by the equivalent

term S/N_oB, where N_o is the equivalent receiver noise power density in watts per hertz and B is the channel bandwidth, as before. This form points out that the received noise power is proportional to the receiver bandwidth; a wide receiver "lets in" more noise than a narrow one.

If we rewrite Shannon's equation to use parameters more commonly used with RF modems, we get:

$$C = B \log_2(1 + (E_b /N_o) \times (C/B))$$

where

E_b = received energy per bit, joules (watt-seconds)

N_o = equivalent received noise density, watts/Hz

The ratio E_b/N_o is the figure of merit for all digital modems; the lower we can make it, the less RF power we'll need to send data at a given rate. Now suppose we have infinite bandwidth available. How low can we make E_b/N_o?

The answer is -1.6 dB, the "Shannon bound." It is theoretically impossible to go lower. But this is still a lot lower than what we see in practice. The best RF binary modems (BPSK) require S/N ratios of at least 10 dB for good performance. Less efficient modems (e.g., FSK) require at least 13-14 dB, and the very worst (AFSK/FM) may require 20 dB or more. And these figures are independent of the receiver's bandwidth; once you have enough bandwidth, more doesn't buy you anything.

Unless you use FEC. This is the key to trading off bandwidth for power. With it we can approach (but not reach) the Shannon bound. In practice, it's fairly easy to take the required E_b /N_o for PSK to 4-5 dB, but 2 dB is about the practical limit with known techniques. Still, this is quite a bit better than the 10 dB required for PSK without FEC.

These gains, called "coding gains," are quite real. If the FEC in use has a coding gain of 5 dB, you can drop your transmitted power (or antenna gain) by 5 dB and still transfer data at the same rate as before! Anyone involved in, say, DXing or moonbounce operation knows that generating large amounts of effective radiated RF power is ex-

pensive and getting more so all the time. FEC reduces the need for RF power, and the computers needed to do FEC are getting cheaper and more powerful all the time. FEC thus represents a real triumph of "brains over brawn."

These figures are based on an important assumption: that all of the noise (and interference, if any) is additive and Gaussian. Additive simply means that the channel (and your receiver front end) is linear. Gaussian implies white-noise, the kind of noise you get from the thermal motion of molecules in a preamp, or (usually) from the cosmic background. But not all noise is Gaussian. On the 70-cm ham band, for example, one important source of noise in many areas is military radar — and the short, high energy pulses from these things are about as non-Gaussian as you can get!

The effect of strong radar QRM is to cause regular bit errors at a rate that depends on the radar's pulse duration and repetition rate and the duration of each data bit. To illustrate the problem, I'll consider a 56-kbit/s Amateur link being strongly QRMed by a type of 70-cm military radar that seems common to both the east and west coasts of the US: a pulse duration of 10-20 microseconds at a pulse rate of about 300-400 Hz. Since the radar pulses last about as long as a single bit at 56 kbit/s (17.9 microseconds to be more precise), simply blanking out the radar pulse won't help. Every 140-186 data bits or so, a radar pulse will take out one (possibly two) data bits. And if the radar pulses are, say, 40-dB stronger than our data signal, the only way to overcome the problem by brute force is to crank up the power by 40 dB-and this may well be impossible.

FEC provides a more elegant way out. Many error-correcting codes can easily handle the relatively low bit-error rate of 0.5-0.7% on our radar-QRMed channel, at some cost in through-put. And for our example of a radar that's 40-dB stronger than our desired signal, this represents a coding gain of 40+ dB! Here the choice is clear: FEC is far more cost effective than increased transmitter power or antenna gain. Indeed, another motivation (besides NASA deep space exploration) behind the early develop-

ment of FEC was the need to protect US Navy shipboard data links from strong radar interference.

FEC is now a standard technique found in many commercial and military radio modems. There are many different FEC codes, about which many textbooks have been written. However, I can summarize them very briefly while introducing the technique I've been experimenting with.

Error-correcting codes fall into two broad categories: block and convolutional. Block codes, as the name implies, work on fixed-sized blocks of data. These work well when the medium is already divided into fixed blocks, such as words in a computer memory or blocks on a magnetic disk, or when the medium is a semi-infinite stream of bits that can be divided into blocks (such as a compact disk). The Hamming code is one block code that's popular in computer memory designs such as those on the MicroSats, and the Reed Solomon code is another (it's used on compact disks). Similar (BCH) codes are used to protect the digital signaling messages in cellular telephone systems.

The other category of FEC, the convolutional code, works with arbitrary amounts of data. This makes it more suitable for variable-sized blocks of data such as those in packet radio, and this is the one I've chosen for my experiments.

Convolutional codes are extremely easy to generate. One feeds the data to be sent down a shift register. Taps at preselected points on the shift register feed networks of exclusive-OR gates. The outputs of these XOR networks are actually sent over the channel. A little thought will show that any given data bit will continue to affect the outputs of the XOR networks until that bit "falls off" the end of the shift register; this length is called the constraint length, K, of the code. The number of XOR networks determines the rate, r, of the code; in a rate 1/2 code (read as "rate one-two"), there are two output bits (produced by two different XOR networks) for each input data bit. Other rates are possible, for example, a rate 3/4 code could be generated by shifting in three data bits and then sending the outputs of four different XOR networks.

The taps to use are determined by the polynomials of the code in use. Many good code polynomials are tabulated in textbooks, so we don't have to find them ourselves.

Generating a convolutional code is the easy part; decoding it is the fun part. The receiver has to determine which bits were sent by observing the outputs of the sender's encoder (with individual bits possibly corrupted by channel noise) and comparing them to a local copy of the encoder. There are two types of algorithms for decoding convolutional codes: parallel and sequential. The parallel (Viterbi) algorithm tries all possible data sequences in parallel, eliminating at each step those that could not possibly be correct. Because of its parallel nature, the Viterbi algorithm is a natural for hardware implementation, and chips are commercially available that will run at multi-megabit speeds.

The sequential algorithm, on the other hand, tries only one sequence of data bits at a time. As long as the sequence being tried seems "reasonable," i.e., its encoded representation faithfully matches what is actually received, the decoder continues forward until it finishes regenerating the sender's entire message. Should a channel error cause the decoder's own copy of the encoder start to diverge from the incoming encoded stream, however, the decoder will "back up" and try other sequences until it finds another one that gets it back on track.

The big advantage of the sequential decoder over the Viterbi decoder is its speed when the channel error rate is low, especially when the constraint length (encoder shift register length, K) is long. The decoder just keeps moving forward, rarely if ever backing up. Indeed, because it must try 2K paths in parallel at all times, Viterbi decoding is impractical when K is more than about 9, but sequential decoding is routinely done with K=32 or 48. On the other hand, sequential decoding "blows up" (fails to make progress) when the error rate becomes very high, while a Viterbi decoder will always decode at the same rate (although it may produce incorrect results).

In general, sequential decoding is preferable when you have to imple-

ment it in software, while Viterbi decoding is the way to go if you have one of the specialized decoder chips available. The variable decoding time of sequential decoding is not a real problem in packet radio since the sender can always time out and resend a packet if the receiver fails to decode it in time.

To see if sequential decoding is a practical technique for Amateur packet radio, I implemented the Fano algorithm (the most popular sequential decoding technique) in 'C' on a 33-MHz 486. I used the same rate 1/2 constraint length, K=32 code that was used on the Pioneer 10 and 11 missions and documented in several textbooks as a "NASA Planetary Standard" code. My decoder ran fast enough to keep up with a channel rate of 56 kbit/s (28 kbit/s user data rate) as long as the channel error rate was below about 2%, well above that required to deal with the military radar QRM figures mentioned earlier.

This demonstrates that with modern microcomputers now available, this 30-year-old algorithm is finally a practical alternative for high-speed Amateur packet radio. Nevertheless, much more work remains to be done before a practical system can be built.

The first requirement is a new packet framing format. Because FEC can decode a packet successfully even when many of its bits are in error, we need a more reliable start-of-frame indication than the HDLC flag. A pseudo-random "sync vector" of, say, 64 bits would do. Such a long sequence could be reliably detected with a reasonably small probability of false alarms, even if a few of the bits are corrupted. This requires a correlator to look for the sync sequence, but this can easily be done in software on the same CPU that does the Fano decoding.

Another practical feature is using FEC only when it is really needed, since this will save fully half of the channel capacity (for a rate 1/2 code). One way to do this starts with a different convolutional code with the "systematic" property — the original data appears in the output as one of the code bits. (You implement this in the encoder by making one of the XOR networks have only one tap on the shift register). Systematic codes perform slightly worse than non-systematic codes, but not by much.

The first transmission of each frame could consist of just the original data, plus a CRC. If no channel errors occur, the receiver will verify the CRC and acknowledge the frame just as it does in conventional packet radio. The sender then goes onto the next frame without sending the additional parity information, and without invoking a decode operation at the receiver.

On the other hand, if the first frame is received with errors, the receiver can signal this fact with a negative acknowledgment (a NAK — as an aside, another advantage of a long sync vector is the ease with which an errored frame can be distinguished from purely random noise). The sender then sends not the original data again, but the first set of parity bits from the convolutional encoder. The receiver combines this frame with the previous version it received and attempts to decode it as a rate 1/2 code. If it succeeds, it finally acknowledges the frame and the sender proceeds to the next frame. If this also fails, a second set of parity bits (different from the first) could be sent and the receiver could attempt a decode of a rate 1/3 code.

Note that successful decoding is possible even if both the original data and parity frames are riddled with errors, as long as the total percentage of bits in error doesn't exceed the error correcting capability of the code. This is in stark contrast to present (noncoded) practice, where the sender must blast the same frame again and again until it finally gets lucky and gets one through without errors. The power of FEC comes from being able to make the maximum use of received information, even when it is contaminated with errors. And since the success of a transmission depends on the error rate, not the absolute total number of errors in the frame, there is no longer a need to keep frames small (and turnaround overhead large) to ensure reasonable throughput. —KA9Q

NK6K here. FEC with ARQ is already in use on the ham bands. HAL's CLOVER HF modem uses Reed-Solomon coding at 60%, 75% or 90% efficiency (ratio of user data to total data). Experimenters interested in FEC at higher speeds and frequencies should contact Phil: karn@unix.ka9q.ampr.org.

5 1/4" Diskettes

TAPR will discontinue distribution of software on 5 1/4" diskettes as of March 1, 1994.

The reasons behind this decision are:

1. Demand for the 5 1/4" diskettes has dropped off.

2. Many of the software packages require multiple 5 1/4" diskettes because of their size, and this trend continues as program sizes increase.

3. Many of the software packages come to us on 3 1/2" media, which means they have to be taken apart and repackaged to fit on multiple 5 1/4" diskettes. An increasing number are packaged with installation programs that were written to work with the 3 1/2" diskettes, making it very difficult to repackage them on 5 1/4" media.

We will evaluate our library and combine some packages now on separate diskettes into one 720k diskette where that is possible, cutting the cost to you, and reducing our diskette inventory as well.

We will provide, on an exception basis, as time permits, software on 5 1/4" diskettes for those of you who cannot handle 3 1/2" media.

After 3/1/94 the cost of 5 1/4" diskettes requested on an exception basis will be $3.00 each, and orders for them will only be accepted via phone or Email so we can advise you about availability and any delays in shipping.

THERE WILL BE NO CHANGE IN THE ORDERING PROCEDURE FOR 3 1/2" DISKETTES AFTER THIS DATE, ONLY THE 5 1/4" DISKETTES WILL HAVE TO BE ORDERED VIA INTERNET MAIL OR TELEPHONE.

This exception service will be terminated at a later date, to be announced in the PSR.

Orders for all diskettes should be placed as they are now, there will be no change in that procedure until 3/1/94.

Software Library Update

by Lou Nigro, KW7H

In addition to supplying various kits and firmware, TAPR maintains a library of packet radio-related computer software. Additions to the software library are always welcome, however we do request that they be submitted either by, or with the expressed permission of, the author. TAPR attempts to provide the latest versions of all software; updates are appreciated. TAPR reserves the right to screen any submissions and restrict the library content as necessary. Both freeware and shareware are acceptable.

Current versions in TAPR software library - As of 14 July 1993. Items with ** notation have been updated since last listing.

Disk No.	Name	Version	Date
1/1A.	APLINK	Ver. 6.04	10-02-92
2/2A	AA4RE BBS	Ver. 2.12	03-31-91
3.	CBBS	Ver. 6.6	03-09-90
4.	EZPAC	Ver. 1.1	01-09-89
5.	MONAX		10-30-87
6.	Ham Comm	Ver. 3.08	03-08-91
7.	TNC-2 Manual and EPROMs		09-29-92**
8.	R95	Ver. 4.0	01-25-92
	7PLUS	Ver. 1.63	01-02-92
	UUENCODE/UUDECODE		01-26-93**
9/9A	ROSERVER PRMBS	Ver. 1.73	05-08-92
10.	ROSE X.25 SWITCH	Ver. 3.1	07-29-92
11/11A	KA9Q NET	Ver. K29	05-18-93**
12.	WXN Weather Svr. Ver. 3.19		04-20-92
13.	TNC1 CODE		05-30-84
14.	TNC2 NOTES		03-28-90
15.	WA7MBL BBS	Ver. 5.14	02-11-90
16.	W0RLI BBS	Ver. 14.2	08-17-92
17.	YAPP	Ver. 2.0	12-18-86
18/18A	INTRO TO TCP/IP		09-09-87
19/19A	LAN-LINK	Ver. 2.01	07-06-92**
20.	ARESDATA	Ver. 1.5	01-20-91
21/21A	MSYS	Ver. 1.13	02-11-92
22.	G8BPQ NODE	Ver. 4.06F	04-25-93**
23.	UTILITIES		
	PKARC	Ver. 3.6	
	PKZIP	Ver. 2.04G	**
	LHA	Ver. 2.11	
	ZOO	Ver. 2.10	**
	UUENCODE/UUDECODE Ver. 5.21		**
24.	THS	Ver. 2.50	11-11-89
25.	VE4UB NTS	Ver. 091891	09-18-91
26.	NM1D DOSGATE	Ver. 1.14	11-29-89
27.	SV7AIZ BBS	Ver. 3.24	04-05-90
28.	TEXNET	Ver. 1.6	02-05-91
29.	INTRO TO PACKET RADIO		11-04-90
30.	MICROSAT GROUND-STATION SOFTWARE		
	PB		10-09-92
	PG		02-25-92
	PFHADD		10-09-92
	PHS		12-21-90
31/31A	No Longer Available (see 38/38A)		
32.	PAMS-Personal AMTOR Mailbox Ver. 2.08		11-26-92
33.	TNC-2 Z-80 Monitor	Ver. 2.00	09-02-91
34.	GIL	Ver. 1.03	03-30-91
35/35A.	PAKET	Ver. 5.1	04-06-92
36/36A/36B.	F6FBB BBS	Ver. 5.15	03-06-93**
37.	TPK	Ver. 164A	04-14-91
38/38A	KA9Q JNOS (Executables, docs.) Ver. 1.08B		
39/39A/39B.	JNOS (Source Code for 38) Ver. 1.08B		
40/40A	SP Packet	Ver. 6.01E	02-06-92**

T **Tucson Amateur Packet Radio**
A P.O. Box 12925 • Tucson, Arizona • 85732
P Office (602) 749-9479 • Fax (602) 749-5636
R TAPR is a Non-Profit Research and Development Corporation

Order Form

Summer 1993

All prices subject to change without notice and are payable in U.S. funds. Prices include shipping and handling except foreign air.
Please allow six to eight weeks for your order to be shipped. For specific information on kits, see Product Description flyer.

Kits and Firmware

Qty	Item	Unit Price	Total	Information
___	TAPR 9600 bps Modem	$ 70.00	___	
___	Bit Regenerator	$ 10.00	___	used for regenerative repeater operation
___	Clock Option	$ 5.00	___	used for regenerative repeater operation
___	Deviation Meter	$ 90.00	___	
___	Trak-Box	$ 195.00	___	limited kits available
___	METCON-1 Telemetry/Control	$ 75.00	___	includes 8 input, 4 output ports
___	4 additional output ports	$ 15.00	___	
___	Voltage-to-Frequency module	$ 25.00	___	
___	Temperature-to-Freq module	$ 35.00	___	
___	A-D Converter	$ 45.00	___	
___	Elapsed Time Pulser	$ 30.00	___	
___	PK-232 Modem Disconnect	$ 20.00	___	simplifies connection of external modems
___	PK232MBX Installation Kit	$ 15.00	___	for installation of 9600 modem in PK-232MBX
___	XR2211 DCD Mod.	$ 15.00	___	
___	State Machine DCD Mod.	$ 15.00	___	
___	State Machine DCD w/Int Clock	$ 20.00	___	For KPC2 or other TNC w/o 16X or 32X int clock
___	PSK Modem	$ 115.00	___	limited kits available
___	Cabinet for PSK Modem	$ 10.00	___	
___	TNC-2 bare PC Board	$ 30.00	___	No parts. Incls schematic, manuals, EPROM code.
___	32K RAM w/ TNC2 update docs	$ 20.00	___	
___	TNC-2 1.1.8a w/KISS EPROM	$ 12.00	___	includes 1.1.8 Commands booklet
___	TNC-2 WA8DED EPROM	$ 10.00	___	8 connect version for ARES/Data standard
___	TNC-1 WA8DED EPROM	$ 10.00	___	
___	PK-87 WA8DED EPROM	$ 10.00	___	
___	TNC-1 KISS EPROM	$ 10.00	___	
___	TNC-2 KISS EPROM	$ 10.00	___	
___	1.1.8 Commands Booklet	$ 5.00	___	full TNC-2 command set for 1.1.8
___	PSR Set Issue #1 to present	$ 30.00	___	
___	TAPR Badge	$ 10.00	___	include Name and Call for badge

Software — please circle disk numbers requested (see PSR or contact the TAPR office for a current list).

___	5-1/4 Disks (each)	$ 2.00	___
___	3-1/2 Disks (each)	$ 3.00	___

1	2/A*	3	4	5	6	7	8	9/A*	10
11/A*	12	13	14	15	16	17	18/A*	19	20
21/A*	22	23	24	25	26	27	28	29	30
31	32	33	34	35/A*	36/AB*	37	38/A*	39/AB**	

* Indicates one 3-1/2" disk. ** Indicates two 3-1/2" disk.
See Separate list for descriptions of these software packages. TAPR attempts to provide the latest versions of all software.

___   ___   **Sub-Total**
         ___   **Arizona Residents add 5% tax**
___   Membership (each year) .... ___   **Membership**
$15 per year US and possessions,   ___   **Airmail Outside U.S.** (Contact TAPR on pricing)
$18 Canada/Mexico, $25 elsewhere   ___   **TOTAL**

Name: _____   Callsign: _____

Address: _____

City, State, Country: _____   Zip: _____

Credit Card Number _____ Expires: _____ Signature: _____
(Visa/Mastercard Only)