$1.75

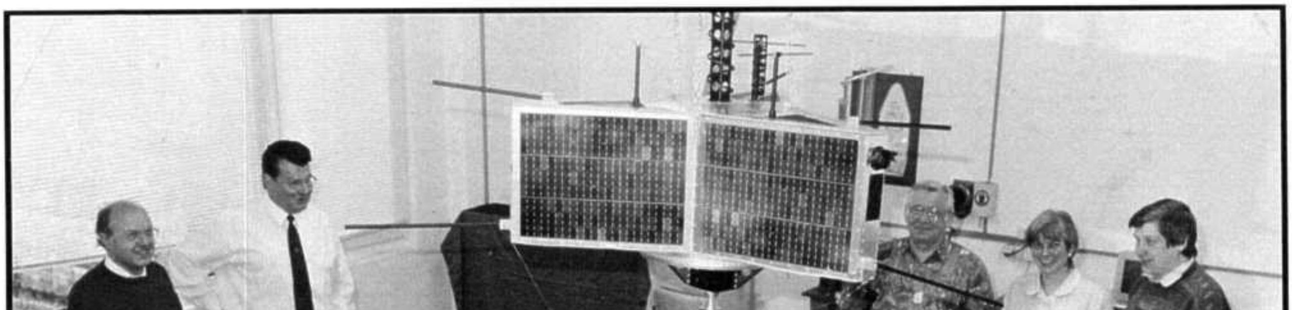
QEX

125

WITH **Gateway**

ARRL Experimenters' Exchange

JULY 1992



X: The ARRL
perimenters' Exchange
nerican Radio Relay League
5 Main Street
wington, CT USA 06111

Non-Profit Org.
US Postage
PAID
Hartford, CT
Permit No. 2929

QEX (ISSN: 0886-8093) is published monthly by the American Radio Relay League, Newington, CT USA.

David Sumner, K1ZZ
Publisher

Paul L. Rinaldo, W4RI
Editor

Lori Weinberg
Assistant Editor

Mark Forbes, KC9C
Geoffrey H. Krauss, WA2GFP
Stan Horzepa, WA1LOU
Contributing Editors

Production Department
Mark J. Wilson, AA2Z
Publications Manager

Michelle Bloom, WB1ENT
Production Supervisor

Sue Fagan
Graphic Design Supervisor

Dianna Roy
Senior Production Assistant

Circulation Department
Debra Jahnke, *Manager*
Kathy Fay, N1GZO, *Deputy Manager*
Cathy Stepina, *QEX Circulation*

Offices
225 Main St, Newington, CT 06111 USA
Telephone: 203-666-1541
Telex: 650215-5052 MCI
FAX: 203-665-7531 (24 hour direct line)
Electronic Mail: MCI MAIL ID: 215-5052
(user name ARRL)
Telemail address: ARRL
Internet:arrl@mcimail.com

Subscription rate for 12 issues:
In the US by Third Class Mail: ARRL Member $12, nonmember $24;

US, Canada and Mexico by First Class Mail:
ARRL Member $25, nonmember $37;

Elsewhere by Airmail:
ARRL Member $48 nonmember $60.

QEX subscription orders, changes of address, and reports of missing or damaged copies may be marked: *QEX* Circulation.

Members are asked to include their membership control number or a label from their *QST* wrapper when applying.

TABLE OF CONTENTS

COLUMNS

On the cover: From left to right, Karl Meinzer, DJ4ZC; Werner Haas, DJ5KQ; Konrad Müller, DG7FDQ; Adjuta Bertsch, Secretariat; and Konrad Schneider. *Photo by Wilfried Gladisch.*

JULY 1992 QEX ADVERTISING INDEX

AMSAT: 24
Communications Specialists Inc: 23
Down East Microwave: 24
Henry Radio Stores: Cov III
Jacob Handwreker, W1FM: 22
King Enterprises: 23
L.L. Grace: Cov II
P.C. Electronics: 15
Schendler Systems, AC4IW: 23
The Modulus People: 22
Yaesu USA Inc: Cov IV

THE AMERICAN RADIO RELAY LEAGUE

The American Radio Relay League, Inc, is a noncommercial association of radio amateurs, organized for the promotion of interests in Amateur Radio communication and experimentation, for the establishment of networks to provide communications in the event of disasters or other emergencies, for the advancement of radio art and of the public welfare, for the representation of the radio amateur in legislative matters, and for the maintenance of fraternalism and a high standard of conduct.

ARRL is an incorporated association without capital stock chartered under the laws of the state of Connecticut, and is an exempt organization under Section 501(c)(3) of the Internal Revenue Code of 1986. Its affairs are governed by a Board of Directors, whose voting members are elected every two years by the general membership. The officers are elected or appointed by the Directors. The League is noncommercial, and no one who could gain financially from the shaping of its affairs is eligible for membership on its Board.

"Of, by, and for the radio amateur," ARRL numbers within its ranks the vast majority of active amateurs in the nation and has a proud history of achievement as the standard-bearer in amateur affairs.

A bona fide interest in Amateur Radio is the only essential qualification of membership; an Amateur Radio license is not a prerequisite, although full voting membership is granted only to licensed amateurs in the US.

Membership inquiries and general correspondence should be addressed to the administrative headquarters at 225 Main Street, Newington, CT 06111 USA.

Telephone: 203-666-1541 Telex: 650215-5052 MCI.
MCI MAIL (electronic mail system) ID: 215-5052
FAX: 203-665-7531 (24-hour direct line)

Canadian membership inquiries and correspondence should be directed to CRRL Headquarters, Box 7009, Station E, London, ON N5Y 4J9, tel 519-660-1200.

Officers

President: GEORGE S. WILSON III, W4OYI
1649 Griffith Ave, Owensboro, KY 42301

Executive Vice President: DAVID SUMNER, K1ZZ

Purpose of *QEX*:

1) provide a medium for the exchange of ideas and information between Amateur Radio experimenters
2) document advanced technical work in the Amateur Radio field
3) support efforts to advance the state of the Amateur Radio art

All correspondence concerning *QEX* should be addressed to the American Radio Relay League, 225 Main Street, Newington, CT 06111 USA. Envelopes containing manuscripts and correspondence for publication in *QEX* should be marked: Editor, *QEX*.

Both theoretical and practical technical articles are welcomed. Manuscripts should be typed and doubled spaced. Please use the standard ARRL abbreviations found in recent editions of *The ARRL Handbook*. Photos should be glossy, black and white positive prints of good definition and contrast, and should be the same size or larger than the size that is to appear in *QEX*.

Empirically Speaking...

Phase 3D Designers Meet in Marburg

Thirty-seven participants were at the AMSAT Phase 3D spacecraft design meeting in Marburg, Germany, May 25-28, 1992. Dr Karl Meinzer, DJ4ZC, chaired the meeting, whose purpose was to take stock of work done by various groups, identify problems, and, where possible, to give some direction to the remaining work.

Phase 3D is to be much larger than previous Phase 3 satellites because of the size of the Ariane 5 launcher and the position of satellite at the bottom of the payload cone. Its size makes it possible to accommodate transponders for every band from 29 MHz through 24 GHz, and that is what AMSAT is trying to achieve. Various groups of experimenters are working on individual transponders, the basic spacecraft, propulsion and internal housekeeping computers. There were interesting presentations made on each of the subsystems. One feature of the spacecraft is a switching matrix to link up any pairs of bands for transmitting and receiving.

Dick Daniels, W4PUJ, had the job of adding up everyone's requests for "real estate," ie, amount of space needed for various boxes mounted on the walls of the satellite. Dick Jansson, WD4FAB, had proposed standard box sizes of 100 × 200, 200 × 200, and 200 × 300-mm, with heights of 25, 32 and 38 mm. Using paper cutouts of these box dimensions, Daniels found that they would not fit on the walls. Meinzer then asked Daniels to total the areas requested and compare them with the total area available: The boxes added up to 50% of the total area. So, if cabling and other incidentals are ignored, there is plenty of room for the boxes but their dimensions and need for specific locations make it an unsolvable puzzle unless some box dimensions are changed.

Ernst Messerschmid, DG2KM, a former Astronaut, who heads the Institute of Space Systems at the University of Stuttgart, introduced his group's project for station-keeping thrusters. Basically, they look like an overgrown garden hose nozzle which, inside, has a spark plug used to ignite gasified ammonia. The Institute has to pay some of the cost of the satellite for the privilege of supplying the thrusters.

In connection with the station-keeping thrusters and the main orbital thrusters, there are to be six fuel tanks mounted in the inside cone of the satellite. Space-qualified tanks cost about $25,000 each, and the whole fuel assembly should run about $175,000. There was some talk about using aluminum beer kegs, but the space qualification tests would offset the savings. Several volunteers were ready to help empty the tanks of their original contents.

Although P3D is complex, the technical part of the project seems to be well within the capabilities of the international AMSAT team. Funding, however, is a different story. As there are many details to be resolved, no one knows the exact cost but estimates are in the neighborhood of $3 million. Nevertheless, it will take contributions from many amateurs worldwide to make P3D a success.

The NAVSTAR Global Positioning System (GPS) had been considered a possible navigation system for P3-D. Matjaz Vidmar, YT3MV, had spent the last year building a GPS receiver—from scratch, using only published specifications. He designed the hardware and built it from parts scrounged on trips to Germany. He wrote all the software. Dr Tom Clark, W3IWI, brought along a commercially built hand-held GPS receiver, whose position location readings were comparable to those produced by Vidmar's box.

The next event in the P3D time line is the AMSAT-UK Colloquium in Surrey, England in late July. Several of the groups were asked to report their progress and proposals at that meeting.

The critical design meeting is to occur in Orlando sometime during November, 1992. Many decisions will have to be made then to meet the launch date.—*W4RI*

The Status and Future of High Frequency Digital Communication

Part II: HF Modems and Their Performance

By Ken Wickwire, KB1JY
232 North Road #17
Bedford, MA 01730

1. Introduction

In the first part of this series I pointed out that modern methods of digital signal processing have played a large part in a rebirth of interest in shortwave radio among commercial, military and amateur users. Most of the signal processing for digital HF takes place in a *modem*. In this part I'll treat in more detail what modems do, illustrate the functions and performance of the types most useful for HF, and touch on the place of modems in the simulation of shortwave digital networks. I'll close with a summary of the modems that are available for amateur operation.

2. What do HF modems do?

The original purpose of a modem gave rise to its name: It was a device for **mod**ulating a carrier with an information signal and **dem**odulating the carrier at the receiver. With the advance of digital signaling, however, the purpose of a modem has gradually expanded to include not only modulation and demodulation, but also a large part of the signal processing needed to transfer data effectively over a channel. For HF radio channels, this additional processing may involve forward error correction (FEC), interleaving, narrow-band interference excision and multipath equalization. If automatic repeat requests (ARQ) are used for error control, this function is sometimes also carried out by software in the modem. (This is the case in AMTOR and for some modems used in the maritime mobile service.) One important reason for the extension of the modem definition is the interplay in a digital system of modulation and error control, which will be discussed below.

One of the most important new developments in digital HF is the use of **automatic link establishment** (ALE). This function is currently carried out by special modems, which have their own modulation and error control. In networks that use ALE, all receivers not already involved in ALE or data transfer periodically scan a set of agreed upon frequencies. When a station wishes to contact another station, it chooses a calling frequency from the set of frequencies according to the available channel quality measurements. (The set of allowable calling frequencies may vary from one pair of stations to another, as it does in Amateur Radio.)

Before transmitting, the calling station listens to hear if the chosen frequency is busy. If the frequency is busy, the calling station tries the next best frequency. If the chosen frequency is not busy, the ALE modem sends ALE calls (containing the addresses of the calling and called stations) for a period of time long enough to allow the scanning receivers to hear them. When a scanning receiver hears its address, it stops scanning and sends a response to the calling station. If the calling station hears the response (usually by recognizing its own and the responding station's address), it sends a confirmation. (In some systems, the beginning of data transfer provides the confirmation.)

Once a link has been established, digital traffic may be sent over it. (So may analog voice, but the emphasis here will be on digital signaling.) Rapid and accurate data transfer is the other main function of HF modems.

In current systems, link establishment and data transfer are carried out by modems in separate enclosures. In the near future, ALE and data transfer will be done by digital signal-processing chips on computer boards, or by one board configured from a terminal to perform both functions, although the signal processing for the two functions may continue to be different. Fig 2-1 illustrates the steps carried out during data transfer over a digital HF link. A comparison with Fig 1-7 of Part I shows that the processing for ALE is generally simpler than that for data transfer. This is because ALE must sometimes be tried several times before data transfer is successful, so its processing should not take too long. This does not mean that ALE processing shouldn't be made as efficient as possible, but merely that it cannot always use techniques like deep interleaving and equalization that take a significant amount of time.

In the next section I'll turn to a general discussion of modulation techniques suitable for use in the changing HF channel. Subsequent sections will treat modulations currently employed in the new US standards for ALE and data transfer.

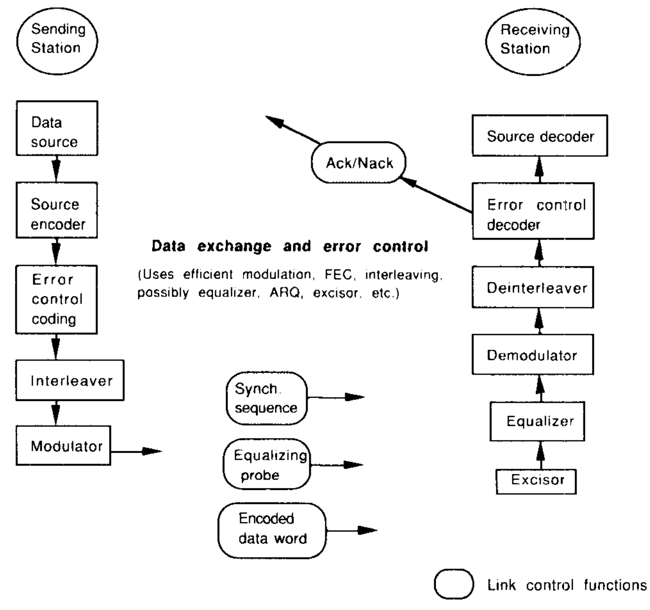


Fig 2-1 Data Transfer (with full signal processing)

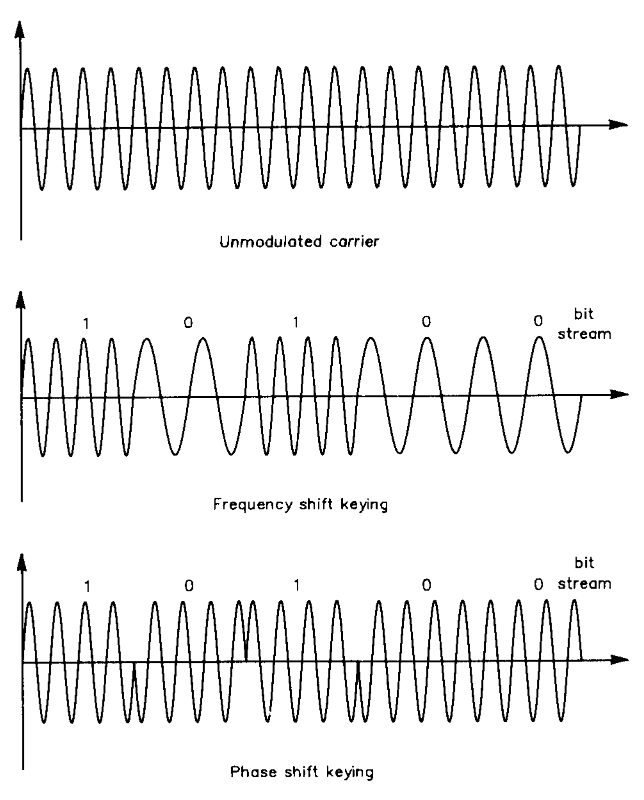


Fig 2-2 Binary FSK and PSK Waveforms

3. Modulations for digital HF

In a modern digital communications system text symbols (eg letters or other characters) of messages to be sent by a station are converted by a **binary source encoder** into a sequence of information bits consisting of zeros and ones. (A much older, *ternary* source code is Morse code, which converts text symbols into dots, dashes and spaces.) An example of a binary source code is the correspondence between zero-one sequences and ASCII characters. If each zero or one is transmitted separately by a different signaling element (eg, frequency tone or phase), we refer to binary modulation. If k bits at a time (called a signaling symbol) are transmitted by $M = 2^k$ different signaling elements, we speak of M-ary modulation. As pointed out in Part I, additional bits are sometimes added to the information bits for the purpose of error control (error detection or correction), especially over the HF channel. We call the combination of information and error control bits *channel bits*. (The modulator and demodulator deal with these channel bits.)

The $M > 1$ signaling elements usually differ in amplitude, phase or frequency. (The elements of Morse code differ in *duration,* and Morse modulation is therefore sometimes called "on-off keying." The usual demodulator for Morse code—the human brain—is still the most accurate demodulator of all, although it is too slow for most applications.) If the absolute phase of a signal is used in its demodulation, we refer to **coherent** demodulation. Otherwise, the demodulation is noncoherent.

Modern digital modulations for HF usually employ changes in the phase or frequency of an RF carrier to produce different signaling elements. In the first case, we refer to the modulation as **phase-shift keying** and in the second as **frequency-shift keying**. Fig 2-2 shows examples of frequency- and phase-shift-keyed waveforms as time functions.

The crowded, noisy nature of the HF channel requires that modulations should, as a rule, occupy small bandwidth and produce small bit error rates. To make transmission efficient, modulation schemes should be linear or have constant envelope. (Constant-envelope modulation allows the use of Class C power amplifiers, which are more efficient than linear amplifiers.) The **symbol duration** of the modulation should be longer than the expected **multipath delay** to prevent intersymbol interference (energy used to send one symbol arrives during the time a different symbol is expected by the demodulator, which raises the error rate). Note that raising the transmitted power cannot reduce intersymbol interference (ISI). An effective, but complicated, way to reduce ISI is to use an equalizer (see Part I and Sec. 4).

The performance of a modulation type or error correction method for a digital system is usually illustrated by giving the corresponding **bit error rate (BER)** curve. This curve shows the average percentage of bits received incorrectly *over a particular channel* as a function of the signal-to-noise ratio (SNR) when the modulation or error correction method is used. The SNR is usually expressed as E_b/N_0, the energy used to send one bit divided by the Gaussian noise level in one Hertz of signaling bandwidth. Using this expression allows signaling or error correction methods to be assessed

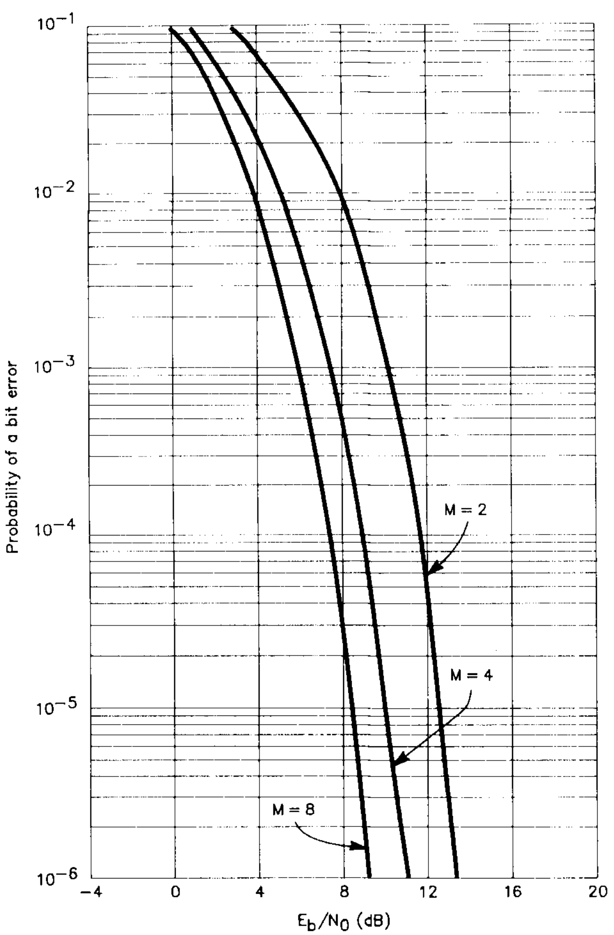


Fig 2-3 Bit Error Probability for M-ary FSK in Additive White Gaussian Noise

independent of signaling bandwidth, which makes comparison easier.

It is customary to give performance for comparison purposes in a channel disturbed only by additive white Gaussian noise (AWGN), even if such a channel is not typical of actual operating conditions. The error rate in AWGN is generally speaking the lowest that can be expected of a modem's performance. However, *the AWGN channel rarely characterizes HF operation.* For HF, performance in channels typical of actual operating conditions is therefore specified in addition to performance in AWGN.

Let's turn now to some modulations that can meet HF requirements.

3.1 Frequency-shift keying (FSK)

Frequency-shift keying uses changes in the RF carrier frequency to produce signaling elements. If the modulator chooses between two frequencies, we speak of **binary frequency-shift-keying (BFSK)**. Conventional teletype, AMTOR and HF packet radio all use BFSK. If M > 2 frequencies are chosen, the modulation is called

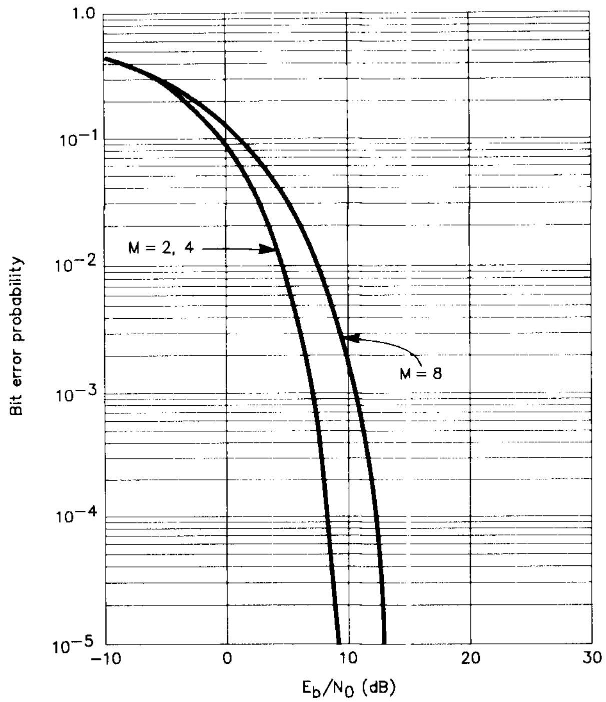


Fig 2-4 Bit Error Probability for M-ary PSK in Additive White Gaussian Noise

M-ary FSK (MFSK). M = 4, 8 or 16 are commonly used. If the frequencies are transmitted one at a time, the modulation is said to be of **serial tone** type (an example of constant envelope modulation). If several frequencies are transmitted at the same time, the modulation is of **parallel tone** type. The current trend in HF is to use serial tone modulation.

Fig 2-3 shows the performance of BFSK modulation in AWGN. Note that nearly error-free communication (BER = 10^{-5}) can be achieved with BFSK in AWGN with a received SNR of about 12 dB. The same figure shows the performance of 4-ary and 8-ary FSK, in which four or eight signaling elements (here, frequencies) are used. These last two FSK modulations yield lower BER in AWGN, because they make more efficient use of the channel than BFSK (they require less energy per bit). This is typical of MFSK in most channels, although the method requires more complicated hardware than binary FSK.

3.2 Phase-shift keying (PSK)

Phase-shift keying uses changes in the phase of a single RF carrier to produce signaling elements. It is sometimes called a serial tone modulation for this reason. Two possible phase changes are referred to as **binary phase-shift keying (BPSK)**; M possible phase changes lead to **MPSK.** Common values for M > 2 are again M = 4 and 8.

Fig 2-4 illustrates the performance of MPSK in

Gaussian noise for M = 2, 4, and 8. Note in this case that a larger signaling set can yield *no change* in BER or even a *higher* BER than a smaller signaling set (see the following remarks). For M = 2, a comparison of Figs 2-3 and 2-4 shows that BPSK is about three dB better than BFSK in terms of bit error rate. (BPSK achieves a BER of 10^{-5} in AWGN at an SNR of about 9 dB.) One reason for this is that signaling elements for BPSK are *negatively* correlated ("bipolar") in Gaussian noise, whereas BFSK elements have only *zero* correlation (they are "orthogonal"); the BPSK elements are thus in a statistical sense farther apart than the BFSK elements, and are therefore easier to distinguish by the demodulator. The price paid for the improved performance with BPSK is more complicated hardware needed to track the phase of the carrier signal.

In the cases of 4-ary and 8-ary PSK, even though the BER does not decrease with increasing M, a larger bit rate can be achieved in the same signaling bandwidth (or the same bit rate can be achieved in a smaller signaling bandwidth). With MFSK, however, increasing M always increases the signaling bandwidth. These facts show that both error and *bandwidth* performance have to be considered in choosing digital modulations for HF.

Remember that the performance of FSK and PSK shown in this section is for AWGN alone, which is not a description of the noise and fading in HF channels. I'll demonstrate modulation performance for typical HF channels in Section 7.

4. Interference excision, equalization and interleaving

These techniques, which I mentioned in Part I, are functions carried out in some advanced HF modems. All three techniques can be viewed as methods for removing most of the HF channel effects, so that the received bit stream *looks like it has passed through AWGN.* Excision removes narrow-band interfering signals (including some "static" caused by lightning) that arrives over the HF channel from far away. Equalization often uses "training signals" to measure the amount of signal distortion caused by multipath, and channel matched filtering methods to restore the distorted signals to their original form. Interleaving spreads the "burst errors" typical of HF fading and static across the bit stream so that they appear to have occurred *independently* (see Part I). Independent or "random" errors are usually easier to correct than burst errors by the error control coding described in the next section.

5. Error detection and correction for digital HF

The history of error control began with the work of **Shannon** in the late forties. Shannon showed that if the **capacity** (measured in bits per second) of an AWGN channel is not exceeded, the bit error rate can be *made as small as one likes* by error correction coding. (The capacity, a function of the channel bandwidth and SNR, can be exceeded if more errors are accepted.) Shannon's result therefore limits the throughput, but not the accuracy of digital communications; if we are willing to add enough "coding" for error control (which reduces the information throughput), we can make the error rate as small as we want, at least in an AWGN channel.

Since the shortwave channel with its burst errors has more structure (or "memory") than the "completely" random AWGN channel, its capacity at the same average SNR in the same bandwidth is actually *greater* than that of the AWGN channel: *if* the structure can be characterized accurately enough, error control schemes tailored to it can in principle be more effective than schemes developed for the Gaussian channel. For example, if we knew that the errors in an HF channel always occurred in groups of seven, we could design a decoding method to detect and correct *all* the errors in the channel. This method would be easier to design and implement than methods used for AWGN.

Unfortunately, errors in the shortwave channel do not occur with such regularity. The impediment to taking advantage of the additional structure of the HF channel to approach its higher capacity is caused by the difficulty of characterizing the channel (where errors occur in groups of variable length), and of tailoring error control schemes to it. Much experimental work on such characterization is needed, and it has just begun.

Since Shannon's work, many mathematicians and engineers have sought effective methods for reducing digital error rates in various channels. As was pointed out in the first part, most methods (excluding the use of more power) involve sending additional **coding bits** with the information bits. The additional bits provide the redundancy (extra information) needed by the receiver to detect, or even correct, errors that occur in the information bits sent over the channel.

The goal of error **correction** coding is to deliver messages with no errors. The goal of error **detection** is to deliver messages with all errors detected, so that other methods (like ARQ) can be used to correct them. Many coding schemes have been developed for detecting and correcting errors in channels of various types, including channels with burst errors.

The most common approach to error control (detection or correction) in the HF channel is **block coding**, which produces **codewords** of n bits. Each codeword contains $k < n$ information bits and $n - k$ coding bits, and the corresponding code is called an (n, k) block code. The **Hamming distance** between two codewords of a block code is the number of places in which they differ. For example, (1010) and (0011) have Hamming distance equal to two. The **minimum distance**, d_{min}, of a block code is the smallest Hamming distance between any pair of its codewords.

It can be shown that a block code can *detect* up to $d_{min} - 1$ errors and *correct* up to $t = [(d_{min} - 1)/2]$ errors, where $[x]$ denotes the largest integer in x. The number k/n, which measures the fraction of the codewords that contains information, is called the **rate** of the code. Most error correcting codes for HF have rates of about 1/2; that is, half of the bits in each codeword are redundant bits used for error correction. In the next sections I'll describe some simple methods of error control. The field is very large and I can only give an outline here.

5.1. Examples of error detection

The simplest method of error detection is a **parity check**. As an example, consider the $(n, n-1)$ code that adds a single bit to the information in a block in such a way that the total number of ones in the resulting codeword is *even*. Thus, (011) becomes (0110), (001) becomes (0011), (010) becomes (0101), etc. For this code, $d_{min} = 2$, so one error can be detected but no errors can be corrected ($t = 0$). Another example is the **AMTOR ARQ error detecting code**, in which each codeword is constructed so that it has exactly four ones and three zeros. The decoder in the AMTOR modem checks to see if the ratio of zeros to ones in each codeword is 3/4, and declares an error if it is not. (A request to repeat the false codeword is then made.)

A further, important example of a detecting code is a **cyclic redundancy check (CRC)**. In a CRC, the bits in an information block and the coding bits are taken as the coefficients of a certain polynomial. The coding bits are added in such a way that when the resulting codeword polynomial is divided by a special, fixed *checking polynomial*, the remainder is zero. The decoder at the receiver performs a similar division and rejects a codeword if the remainder it gets is not zero. CRCs can detect many error patterns in codewords and are used in several international communication standards. Very few false codewords escape detection by a properly chosen CRC.

An early and still important class of detecting codes are the $(2^m - 1, 2^m - 1 - m)$ **Hamming codes**. The added coding or parity bits in Hamming codes are sums of pairs of the information bits at the beginning of each codeword. For the Hamming codes, $d_{min} = 3$, so these codes can detect up to two errors in a codeword. As an example, consider the (7, 4) Hamming code. For this code it can be shown that for binary signaling over a channel with uncoded BER of p, the probability of an undetected bit error after decoding is approximately $7p^3(1-p)^4$. For $p = 0.01$, the probability of an undetected error in this case is only about 10^{-5}.

5.2. Examples of error correction

A simple example of an error correcting code is an $(n, 1)$ **binary repetition code**, in which each bit is repeated n times. That is, n coding bits are added to each information bit to form a codeword. Thus, a one is coded as (1111...1) and a zero as (0000...0). Such a code has $d_{min} = n$ and can thus correct up to $[(n-1)/2]$ errors through a simple majority voting technique if the majority of the received bits are the same as the information bit. As an example, consider the binary channel referred to above (which is only an approximation of a shortwave channel), and suppose that the probability of a bit error is $p = 0.001$. Then the probability of an error *after* decoding of a (3, 1) repetition code is $p^3 + 3p^2(1-p) \approx 3 \times 10^{-6}$, a reduction of about three orders of magnitude. (3, 1) repetition is used in the new federal standard for ALE.

More efficient codes use powerful methods of linear algebra over so-called binary number fields to locate errors among the information bits in a codeword. Once the wrong bits are located, they are corrected by simple binary addition of a one to the wrong bit. (The description of error correction methods in general is beyond the scope of this introductory treatment.)

The Hamming codes are one of the earliest examples of error correcting codes. Since $d_{min} = 3$ for these codes, they can correct all codewords with exactly one error. Codewords with two or more errors (common in HF) cannot be corrected by the Hamming codes, so these codes are not generally used in HF work. (As we saw above, the Hamming codes can also be used to *detect* up to two errors in a codeword. However, they cannot be used to detect and correct errors simultaneously.)

The Hamming codes are **perfect codes**, which are very rare. (The wasteful binary repetition codes with n odd are also perfect.) Perfect codes have the property that their codewords fit "perfectly" into the "space" of codewords: every received codeword is at most a (Hamming) distance $t = [(d_{min} - 1)/2]$ from a possible transmitted codeword, and all patterns of t errors in codewords can be corrected. Perfect codes have the least redundancy necessary for their required error control task (detection or correction, and sometimes both).

The only other example of a perfect code is the (23, 12) **Golay code**, which has $d_{min} = 7$ and can thus detect up to six codeword errors and correct up to three errors. Codes such as the Golay with large minimum distances can be used to correct some errors in codewords and merely detect others. This possibility is exploited in so-called hybrid error control methods (see below). The Golay code, which can correct randomly occurring errors at channel error rates up to about 12 %, is used in several commercial HF modems for ALE or data transfer (see the discussion below).

Other important topics in error control for HF are codes for correcting burst errors, codes that handle symbols rather than bits (non-binary codes), **code combining** (discussed below) and **convolutional codes**, which deal with continuous bit streams rather than blocks of data.

Fig 2-5 shows the effect of using Hamming and Golay codes to control errors in an AWGN channel with BPSK modulation. (The (24, 12) Golay code is a slight modification of the (23, 12) Golay code used when 24-bit codewords are desirable.) Note that the Golay code can lower the BER achieved by the Hamming code by almost two orders of magnitude at moderate SNRs. This is accomplished at the cost of lower throughput.

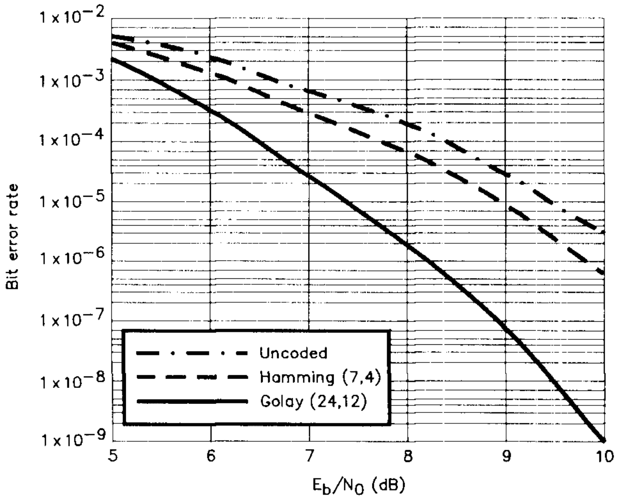


Fig 2-5 Coding with BPSK Modulation in AWGN (hard decisions)

5.3. Hard and soft decoding

There are two basic ways for a receiver to decide whether a binary bit is a zero or one. One way is to let the demodulator make the decision using a threshold criterion. This approach uses only two "quantization levels" to describe the incoming signal (the signal energy is "above threshold or below threshold") and is called **hard decision decoding**. Another approach is to use more than two signal quantization levels as the output from the demodulator and let the decoder make the decision using the quantized levels. This approach, which uses more information than hard decision decoding, is called **soft decision decoding**. As a general rule, soft decision decoding requires about three dB less signal energy to produce the same average bit error rate as hard decoding. As usual, the improvement is achieved at the cost of more complicated and more expensive hardware. Soft decision decoding is used in some commercial modems, but like FEC itself, it has not yet become cheap enough for amateur use. (This will change in the near future.)

5.4. Automatic repeat request (ARQ)

The HF channel can usually be accurately described as one with a combination of single, independent ("random") errors caused by Gaussian noise and occasional strings of burst errors caused by fading, lightning and radio interference. Most of the time the error correction techniques discussed above (including interleaving) can correct the random errors. Sometimes, however, a data packet will come from the decoder in error: its errors will usually be detected but not corrected. ARQ techniques provide a way for the receiver to inform the transmitter that it has received incorrect packets and ask to have them sent again. Since it is important that very few erroneous packets go undetected, the error detection codes associated with ARQ are usually very powerful (for example, cyclic redundancy codes).

ARQ approaches in general can be divided into two categories: **continuous** and **discontinuous.** Continuous ARQ is common on full-duplex links, as with satellites or telephone lines, whereas discontinuous ARQ is appropriate for the half-duplex links usually operated in shortwave radio. In both approaches, the message information to be sent is divided into groups of bits called **frames.**

In continuous ARQ, frames are numbered and sent in a steady stream. The receiver notifies the sender over the return link (with a NACK) of the numbers of frames received in error, which are then repeated as part of the stream.

In discontinuous ARQ, a fixed number of frames is sent in a **window.** The size of the window is the number of frames it contains. (The maximum window size allowed in the AX.25 amateur packet protocol is seven frames.) The transmitter waits for an acknowledgment (ACK) or negative acknowledgment (NACK) of the frames in the window before sending another window. When the window contains just one frame, the technique is called **Stop and Wait** ARQ. This is the error control scheme commonly used in amateur HF packet radio.

Two other methods of discontinuous ARQ have been used in, or studied for, HF. One is called **Go Back (to the first frame in error)**, in which an erroneous frame and all frames in the window transmitted after it are rejected and then retransmitted. In the other discontinuous method, called **Selective Repeat**, only the erroneous frames are rejected and retransmitted. (Versions of these two kinds of ARQ are also used on full-duplex links, where they are considered as continuous methods.)

When the channel causes few errors, Stop and Wait ARQ is simple and effective. However, when errors are frequent, Stop and Wait is very inefficient (throughput is very low) because of the large numbers of NACKs. In such cases, Go Back and Selective Repeat are better. It can be shown that Selective Repeat is the most efficient of the three ARQ approaches. However, it requires more buffer storage and more complicated software to keep track of accepted and rejected frames.

The throughput that can be achieved with all three techniques on half-duplex links is a function of the frame size, the window size and the bit or burst error rate in the channel. In more advanced adaptive techniques now being studied for HF, the window and frame size are adjusted as the bit error rate in the channel changes. When errors are very frequent—as in networks with a lot of contention—ARQ techniques may lead to very low throughput because they use up too much time sending repeats. This phenomenon is well known in some amateur HF packet networks. In this case, other methods such as using higher power (not usually recommended) or a more orderly channel access scheme may improve performance.

Phil Karn, KA9Q, has proposed a modification of the Stop and Wait scheme, called **ACK-ACK,** which increases efficiency when contention or poor channel conditions lead to many repeats caused by lost acknowledgments. In Karn's scheme, which is based on ideas first proposed by the designers of the ALOHANET, frames (packets) are numbered for identification, and the receiver sets a timer when it sends each ACK. If the transmitter has not acknowledged the ACK (with an "ACK-ACK" or a new packet) before the timer expires, the receiver sends another ACK. Since new packets serve as ACK-ACKs so long as ACKs are not lost, ACK-ACKs need only be sent after lost ACKs and at the end of the traffic, so the channel is not occupied unnecessarily. Karn has quoted simulation studies that suggest much higher efficiency for the ACK-ACK protocol than for the standard Stop and Wait protocol, and has pointed out the additional advantages that can come from adaptive adjustment of the parameters (frame size, time-out interval, etc) of the ACK-ACK protocol.

I've just touched on the most common of the many approaches available for controlling errors in the HF channel. Other approaches that should be mentioned are **hybrid ARQ** and **code combining**. Both methods are based on the idea that a combination of FEC and ARQ (for those who can afford it) is the most effective way to reduce errors. They use codes with large minimum distances that can correct some errors and detect the rest, or a combination of a correcting code and a detecting code.

Hybrid ARQ can be used when both low delivery time and low error rate are important. It offers the reliability of ARQ without loss of capacity when the BER is high. In this technique, coding overhead bits are used for both correction and detection. Codewords are first passed through a bit error corrector, which corrects some errors and lowers the error rate. The codewords then pass through a bit error detector, which uses the rest of the overhead bits to detect any remaining errors. If necessary, a request is then sent to the transmitter to repeat erroneous frames.

Code combining is used in systems that must deliver very accurate data, possibly at the expense of large delivery times. In code combining, codewords are also first passed through a corrector and then a detector. If errors remain, a NACK is sent and the erroneous frames are sent again *at a lower code rate*—typically half the previous rate. The retransmitted frames again go through correction and detection, and correct frames are combined with the previous correct frames. This continues until the whole message has been received correctly, or until time-out.

I should mention that ARQ and long interleaving are not generally suitable for use with digitized voice traffic because of the unacceptable delays between words they cause.

6. Further remarks on HF channels from the modem perspective

I pointed out above that additive white Gaussian noise alone is not an accurate description of the effect on signals of the HF channel, which is much more complicated than the time-invariant AWGN channel. An accurate characterization of the channel is important for deciding what kind of modem will be needed to operate in it. In addition to noise and interference in our channel, the phenomena discussed in Part I lead to various combinations of AWGN *plus*

- Rayleigh fading
- selective fading
- discrete multipath
- "deep fading"
- "flutter fading," and
- Doppler shifts.

Rayleigh fading is caused by (among other things) the presence of moving refracting or scattering regions in the ionosphere, which cause a transmitted carrier to arrive at a receiver as a sum of multipath components with random, time-varying phases. The envelope (magnitude) of such a sum has a so-called Rayleigh probability distribution. Shortwave signals subject to Rayleigh fading usually exhibit relatively slow variations in signal strength that extend across the entire signal bandwidth. That is, the fading is **nonselective.**

Fig 2-6 shows the BER performance of BFSK and BPSK in a Rayleigh fading channel as functions of the *average* SNR. A comparison with Figs 2-3 and 2-4 shows that whereas the BER with these modulations in AWGN alone falls exponentially with increasing SNR, in Rayleigh fading it falls only as the reciprocal of the average SNR. Thus, Rayleigh fading can cause a serious lowering of performance as measured by BER. (In the notes at the end of this part is an illustration of the difference between a "random" Gaussian channel and an HF channel with

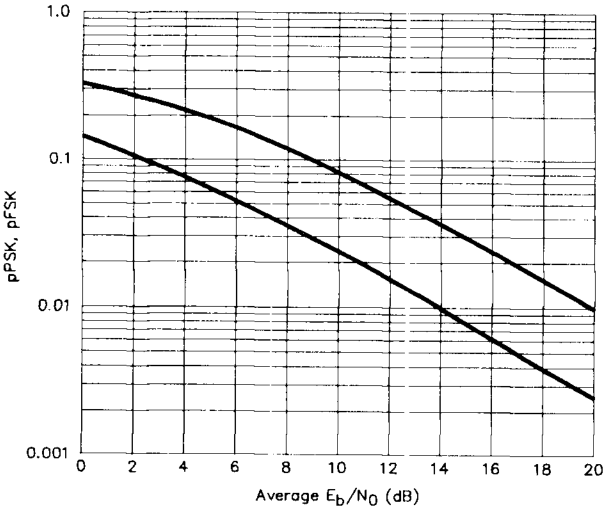


Fig 2-6 Performance of BFSK (upper curve) and BPSK in Rayleigh Fading (BER ~ 1/average SNR).

Rayleigh fading, as measured by the distribution of the gaps between errors in the two channels.)

One method of avoiding this drop in performance is to use **diversity techniques**, which involve sending the signal more than once, setting up multiple receiving antennas, or using FEC codes with large d_{min}. The idea behind diversity is to send "copies" of the information that are not all affected by fades. Another approach is to use **adaptive control of the data rate**, lowering the rate when the SNR drops below the level required for the desired BER and raising it when a fade ends. (This requires feedback from the receiving station.) Further improvements in performance can be gained with equalization and interleaving. All these methods bring the BER curves for Rayleigh fading down into the region where they again fall as exp(*–SNR*). The price paid for this is usually greater delays.

Rayleigh fading is described in purely statistical terms and is often viewed as a phenomenon caused by the combination of a large number of random signal components. For some modem tests it is desirable to simulate multipath caused by a small number (usually two) of **discrete multipath** components that arrive at the receiver with a fixed difference in arrival times. This difference in arrival times is sometimes called the **delay spread** of the components. Discrete multipath can lead to *non-Rayleigh fading* and is best analyzed with an RF channel simulator (see below). The International Radio Consultative Committee (CCIR) has specified several standard discrete multipath channels for the testing of HF modems. These standard HF digital channels will also be touched on in Part III.

If the changes in the ionosphere are such that different frequency components *within* the signaling bandwidth are affected differently by the channel, then the signal may be seriously distorted, and we speak of **selective fading**. Diversity techniques are also used to combat selective fading.

Flutter fading is sometimes experienced on very short links (especially at sunrise) or links near the *aurora borealis* (northern lights) where a large number of rapidly changing multipath components cause the received signal to experience very rapid fading ("fluttering").

Fading in the HF channel causes burst errors. These can cause amplitude variations that are not Rayleigh distributed. If burst errors are relatively rare, ARQ applied to messages sent in the packet mode may provide acceptable performance. Another method, the use of a single FEC code that can control both burst and isolated errors, is often inefficient since the large numbers of overhead bits required are only rarely needed.

Deep fades, as considered here, are not generally associated with multipath but with changes in the ionosphere, such as those caused by sudden ionospheric disturbances (see Part I). Such changes cause the signal strength to fall because most of the signal is no longer being refracted to the receiving site. These fades are usually very brief, but they occasionally last for hours. There is nothing a modem can do to counter a long period of deep fading. Such fading can sometimes be dealt with by using widely separated antennas, or a *relay* that lies outside the region affected by the fade.

The last major disturbance in the HF channel that is directly related to modem design is caused by regular motion of the refracting part of the ionosphere, or by movement of a transmitting of receiving platform (such as an airplane). These movements change the received carrier frequency by a constant amount, called a **Doppler shift**, which can cause a modem to fall out of synchronization or experience large amounts of intersymbol interference. An advanced modem can often track the Doppler shift and compensate (adjust the shifted carrier frequency) for it.

During the design and development of an HF modem it has become customary to test it with a **real-time channel simulator**, a piece of hardware that can simulate RF signals in a number of **standard channels**. Although a channel simulator cannot reproduce all of the effects of a real HF channel, it can generate channels that may be too complicated to model mathematically. RF channel simulators offer the additional advantage that they allow one to test the operation of all the modem's hardware rather than just its signal processing software. Several standard channels have been agreed upon by the CCIR. The standard channels are AWGN and several combinations of Rayleigh fading and discrete multipath delay; the faster the Rayleigh fading and the greater the delay, the worse the channel.

RF channel simulators do not simulate deep fades caused by ionospheric blackouts, noise bursts or radio interference. These are currently dealt with in a network simulation along the lines to be described in Part III.

7. Performance and throughput of HF modems

The performance of ALE modems is given by the **probability of linking** versus SNR in a particular HF channel. This probability is either estimated from tests with an RF channel simulator or derived analytically from the ALE modem's bit error rate (BER). Data modems are assessed by giving their **BER versus SNR** in a particular channel. If data modems use equalizers, the probability and average time of modem synchronization is also given. The particular data word and data frame structure used in a network simulation may require conversion of bit error rates to word error rates or frame error rates, but the starting point for data modem performance specification is the modem's BER. When a simulation is used for HF network design, two of its several inputs (see Part III) are the linking probability of ALE modems and the BER or word error rate of data modems in various types of shortwave channels.

The information rate—in information bits per second (bit/s)—*achieved* by a pair of HF modems operating in an actual channel (the achieved **throughput**) depends on how much fading, noise and radio interference is in the channel. Because of the need to send ACKs and repeat packets that are received in error, this rate is generally lower than the rate at which the transmitting modem *sends* information.

Amateur HF modems, which do not yet use any error control other than stop-and-wait ARQ, normally send information at about 300 bit/s. Their achieved *link* throughput is usually somewhere between 50 and 150 bit/s, but it can be much lower in bad channels.

Commercial HF modems send information at rates between about 100 bit/s for conventional radioteletype modems and 2400 bit/s for advanced modems with FEC, interleavers and equalizers. Their throughput in real channels is also lower than their sending rates. Typical values of achieved throughput lie between a few bits per second for very poor channels and about 600 or 1200 bit/s in quiet (nearly Gaussian) channels. In a network with relays, average throughput is generally lower than the throughput on an individual link, but the point of using automatic relays is that messages eventually get delivered that would not be delivered at all without relaying. In a well designed automatic network operating in quiet channels, the average throughput might be several hundred bits per second.

8. Modems developed according to the new US standards

8.1. ALE modems (Federal Standard 1045)

The first step in standards development dealt with **Automatic Link Establishment**. In 1990, after some months of testing, the Institute for Telecommunications Sciences of the US Department of Commerce produced Federal Standard 1045, the first of the new standards for digital HF networking. The 1045 standard prescribes protocols for automatic linking with individual stations, or with several stations at the same time ("net" calls).

Federal Standard 1045 prescribes an efficient 8-ary FSK waveform for linking (see Section 3.1 above) along with powerful FEC (a combination of Golay coding, interleaving and three-fold word repetition with a majority voting technique). It also spells out the timing and other details of the three-way ALE exchange that must be followed in any ALE equipment purchased for governmental use. Fig 2-7 shows the structure of the frames sent during the ALE exchange. The linking protocol is *asynchronous:* a station may attempt ALE at any time after listening to make sure the chosen frequency is not occupied. Channel quality is to be measured by SNR or bit error rate. These measurements can be made passively or actively, as part of a scheduled sounding or polling scheme (see Pt. I and Sec. 8.3 below). In addition to its powerful FEC, the standard prescribes "command modes" that provide a framework for the CRC checks and NACKs needed for an ARQ error control scheme. Finally, although it deals mainly with ALE, 1045 also includes a provision for the exchange of short data messages.

To meet the standard, ALE modems must achieve prescribed *probabilities of linking* in AWGN and in two simulated CCIR channels that cause Rayleigh fading and discrete multipath components. Several US manufacturers are now producing ALE modems that conform to the basic requirements of "FED-STD-1045," and a series of on-the-air tests is underway to provide data on how the new equipment performs.

To get an idea of the operation and performance of a network using FED-STD-1045 ALE modems consider the spreadsheet excerpt shown on page 10. It contains data from some recent over-the-air tests of a network of ALE stations that are running around the clock on ten frequencies spread more or less evenly from 2 to 27 MHz. The stations are using 100-watt transmitters and various kinds of antennas. The spreadsheet was produced from data-logging software installed in a PC connected to an ALE modem at one of the stations near Boston (MIB).

All significant operations of the modem (ALE call placements and receptions, one-way soundings, two-way "link-quality assessments (LQAs)," and so on were registered by the PC after being tagged with date and time. The other stations connected with this particular page of the spreadsheet are in Virginia (MTR), Illinois (MOT) and Florida (SUN). All the entries without a second address refer to soundings received at Boston. Soundings are attempted on all ten frequencies, so a frequency missing from a particular sequence of soundings is one whose signal was not strong enough to be detected at Boston.

The first eight entries list soundings from 9 to 27 MHz from Illinois and Florida at about two in the afternoon local time. The short-term channel quality on these frequencies is high, as reflected by the SINADs and BERs. (The SINADs and BERs are measured by different processes and are thus not connected by a simple relationship, and blank entries mean that no quality data were received.) There follows a two-way LQA exchange with Virginia initiated by Boston. Only three of the ten frequencies (9, 10 and 13 MHz) were good enough to allow the two stations to connect and measure channel quality.

Then come a pair of individual calls (using the three-way handshake) from Virginia to Boston on 9 MHz. Virginia has used the just measured channel measurements to choose the frequencies it will try for these calls. Comparison of the quality measured during the handshakes with the quality recorded during the LQA exchange eight minutes earlier gives an idea of how

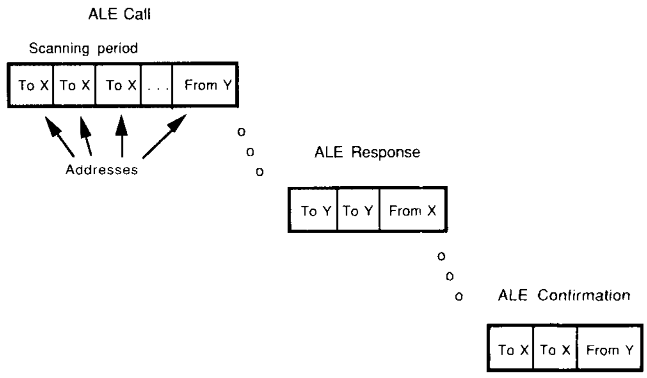


Fig 2-7 ALE Frame Structure

Date	GMT	ALE Event	To/	To/	Ch.	Freq.	Mode	Rcvd	Meas.	Rcvd	Meas.
			From	From	No.	(MHz)		SINAD	SINAD	BER	BER
1/29/92	19:08:38	Sound_received___from	MOT		4	9.97	USB	--	18	-.----	0
1/29/92	19:37:07	Sound_received___from	SUN		4	9.97	USB	--	10	-.----	0.083
1/29/92	19:37:14	Sound_received___from	SUN		5	10.4	USB	--	17	-.----	0
1/29/92	19:37:22	Sound_received___from	SUN		7	15.7	USB	--	30	-.----	0
1/29/92	19:37:29	Sound_received___from	SUN		8	18.2	USB	--	30	-.----	0
1/29/92	19:37:36	Sound_received___from	SUN		9	23	USB		30		0
1/29/92	19:37:48	Sound_received___from	SUN		10	27.5	USB	--	30	-.----	0
1/29/92	20:33:35	Sound_received___from	MOT		9	23	USB	--	20	-.----	0
1/29/92	20:43:36	Lqa_initiated_to__add	MTR	MIB	10						
1/29/92	20:44:12	Lqa_initiated_to__add	MTR	MIB	9						
1/29/92	20:44:47	Lqa_initiated_to__add	MTR	MIB	8						
1/29/92	20:45:23	Lqa_initiated_to__add	MTR	MIB	7						
1/29/92	20:45:39	Lqa_initiated_to__add	MTR	MIB	6	13.4	USB	--	30	0	0.007
1/29/92	20:45:55	Lqa_initiated_to__add	MTR	MIB	5	10.4	USB	--	16	0	0.007
1/29/92	20:46:10	Lqa_initiated_to__add	MTR	MIB	4	9.97	USB	--	30	0	0
1/29/92	20:46:46	Lqa_initiated_to__add	MTR	MIB	3						
1/29/92	20:47:21	Lqa_initiated_to__add	MTR	MIB	2						
1/29/92	20:47:57	Lqa_initiated_to__add	MTR	MIB	1						
1/29/92	20:53:49	IND_Call_received_from	MTR	MIB	4	9.97	USB	--	28	0	0
1/29/92	20:54:06	IND_Call_received_from	MTR	MIB	4	9.97	USB	--	30	0	0
1/29/92	20:57:35	Sound_received___from	MOT		8	18.2	USB		20		0
1/29/92	21:06:43	Sound_received___from	SUN		4	9.97	USB	--	10	-.----	0.051
1/29/92	21:06:51	Sound_received___from	SUN		5	10.4	USB	--	30	-.----	0
1/29/92	21:07:03	Sound_received___from	SUN		7	15.7	USB	--	30	-.----	0.014
1/29/92	21:07:10	Sound_received___from	SUN		8	18.2	USB	--	30	-.----	0
1/29/92	21:07:17	Sound_received___from	SUN		9	23	USB		30		0
1/29/92	21:07:24	Sound_received___from	SUN		10	27.5	USB	--	30	-.----	0
1/29/92	21:21:37	Sound_received___from	MOT		7	15.7	USB	--	16	-.----	0
1/29/92	21:45:37	Sound_received___from	MOT		6	13.4	USB	--	18	-.----	0
1/29/92	21:45:43	Sound_received___from	MOT		6	13.4	USB	--	16	-.----	0.014
1/29/92	22:09:43	Sound_received___from	MOT		5	10.4	USB	--	13	-.----	0.029

rapidly the channel was changing at this time.

Then come series of soundings from Chicago and Florida. The corresponding links have moderate to good quality, and comparison with the soundings made two hours before shows how fast the channel was changing in the medium term.

Other records show unexpected changes in propagation at various frequencies caused by the phenomena described in Part I. In these cases, the 1045 ALE equipment will use stored channel quality data of the type shown here to find and automatically link on alternative frequencies if they can be found in the set of frequencies the net is using.

8.2 Data modems (Military Standard 110A)

National and international standards for HF data modems have existed for several decades, since the introduction of HF radioteletype (telex) in the thirties. In the last ten years, these standards have been extended to include more advanced waveforms like MFSK and parallel-tone formats. Recently the US standards have been further extended to include a very efficient serial-tone PSK waveform suitable for use in disturbed HF channels. This waveform, along with BFSK at various shifts (for various data rates), and two parallel-tone formats, are embodied in MIL-STD-110A. Several of these waveforms are also connected in the standard with forward error correction. MIL-STD-110A modems have recently been built on cards for PC insertion by several manufacturers, and their prices (now in the thousands) are likely to fall as competition increases.

8.3. A brief diversion on frame structure

Data are usually sent by a terminal to a modem in the form of **frames.** A frame consists of a group of bits with markers to show where the frame begins and ends. The bits are organized into special sections for addresses, message priority, message information and error control bits. Common nowadays are *bit-oriented* framing protocols. (Older digital systems used character-oriented framing protocols.) In a bit-oriented protocol, address, data and other fields form a continuous bit stream, and frames are separated from each other ("delimited") by special bit sequences called *frame delimiter flags.* All other bit sequences in a frame are modified if necessary by adding extra "stuffing" bits to make them different from flag sequences. The first part of a data frame often contains special addressing and control information and is called a **header** (the header in an amateur packet contains call signs, for example).

Fig 2-8 shows the structure of a typical message frame that might be used in digital radio communication. A standard bit-oriented framing protocol used for digital communications is the *High Level Data Link (HDLC)* protocol, which forms the basis of the framing protocol used in amateur packet radio. The structure of HDLC frames is illustrated in Fig 2-9. The *supervisory* and *unnumbered* frames shown in the figure are used for ARQ, flow control, polling and other functions. HDLC allows both the Go Back and Selective Repeat ARQ techniques.

As a general rule, frame (or packet) formatting for data transfer with an HF modem is performed by a computer that controls the modem and radio at each node. A noted exception to this practice is the packet assembly and disassembly (PAD) carried out by an amateur HF packet radio modem.

8.4. Other HF standards related to modem operation

The US standards groups are now working on draft standards for **sounding** and **polling** using the basic ALE waveform prescribed in FED-STD-1045. These techniques, which were defined in Part I, are used by one or more stations to find out which other stations in a network are able to communicate, and how good the channels to them are. Further work is underway on draft standards for **exchange of connectivity information** and for **measurement of channel quality** (see Parts III and IV). These developments will be accompanied by extensive experimental tests with RF channel simulators, network simulations and on-the-air tests. (On-the-air tests using the new ALE standard will no doubt continue during the entire course of the standards development.)

8.5. On-the-air testing

As the standards work proceeds, the new standards will treat techniques that are much more complicated than ALE, and about which little is now known (for example, exchange of connectivity information). To write useful standards for using these techniques it will be necessary to calibrate simulation results with *actual* network performance. Networks available for such calibration will have to be flexible enough to allow rapid changes in their control software and efficient collection and analysis of performance data. It is desirable that Amateur Radio operators contribute to these developments.

9. A summary of modems available for amateur work

Modems for amateur digital communication in the HF band are (or have been) available in the US from:

- Amateur Electronic Applications (PK-87, PK-88, PK-232MBX, DSP-2232, DSP-1232 and HFM-64),
- Heath (HK-232 and HD-4040),
- Kantronics (KPC-1, KPC-2, KPC-2400, UTU-XT/P and KAM)

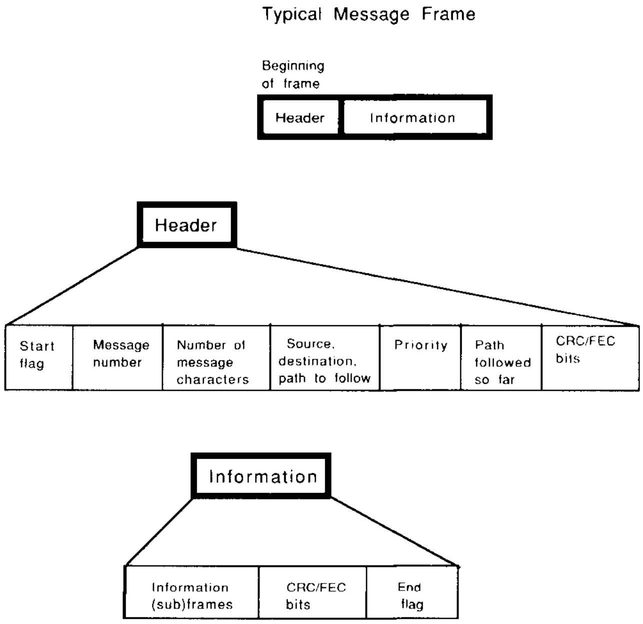


Fig 2-8 A Typical Digital Message Frame and its Contents

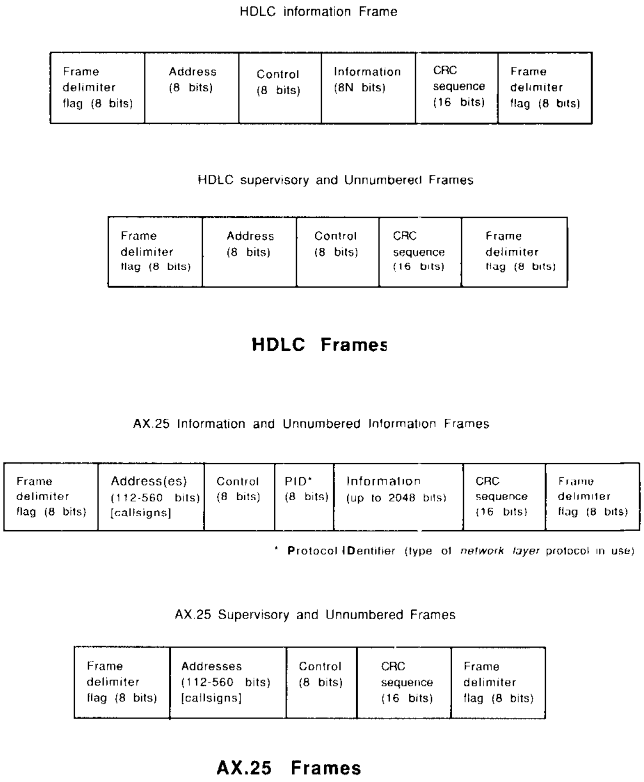


Fig 2-9 HDLC and AX.25 Information and Control Frames

- MFJ (1270B, 1274 and 1278) and
- PacComm (TNC-220, TNC-320, PC-110, PC-120 and PC-320), among others

Many of these are based on the standard Bell 103 BFSK telephone modem, and operate at information rates of 300 bit/s at HF. The shift between FSK (or AFSK) tones is usually set at roughly the information rate (100-300 Hz) to keep the tones orthogonal and reduce filter noise and frequency-selective fading. (Some products also have tuning indicators and allow filters to be adjusted to improve HF performance.)

- HAL Communications Corporation has recently announced its PCI-3000 multimode controller card (for CW, RTTY and AMTOR) that can be mounted in a full-sized IBM PC slot. The company also intends to market the advanced CLOVER modem described below.

Many of these products come with software for operation according to the most recent version of the AX.25 packet radio protocol, which uses a CRC for error detection and a Stop and Wait ARQ protocol, with the option of extending the frame window size up to seven (if any frame is erroneous, all frames in the window are repeated). Software is available for controlling most of these modems from IBM, Macintosh, Commodore and other personal computers.

A few of the newest modems (from AEA and HAL, for example) offer PSK modulation and contain digital signal processing (DSP) chips that can be programmed from a personal computer with modulations and other processing of the user's choice. The multi-mode controllers usually provide AMTOR, RTTY, ASCII, CW and sometimes other digital modes. Several products contain software for NET/ROM packet routing (see Part IV). With the exception of the CLOVER modem, no product listed above uses FEC, interleaving, equalization or interference excision.

- A new modem developed for amateur use by Ray Petit, W7GHM, is called CLOVER. It combines advanced modulation with FEC and an adjustable data rate. The CLOVER modem allows the user to choose modulations consisting of various bandwidth-efficient combinations of *discrete amplitudes* and phases (called **quadrature amplitude modulation**, or QAM). Channel rates (for coding and information bits) between 31.25 and 750 bit/s (with data rates up to about 500 bit/s) may be selected, depending on the channel quality. The CLOVER FEC is provided by the Reed-Solomon code, a powerful, nonbinary (symbol) code that can correct burst errors. ARQ with selective repeats has also been used successfully with the CLOVER on an experimental basis. HAL Communications has announced plans to manufacture this modem, which shows promise of effective performance in poor channels and high throughput in good ones.

Martin Clas, DL1ZAM and Peter Mack, DL3FCJ have recently developed an improved, experimental, version of AMTOR called *PACTOR,* which is meant to raise throughput when channel conditions are good by reducing overhead, and to lower the number of errors when conditions are bad. These are accomplished by the use of

- longer ARQ frames (twelve instead of four bits),
- a 16-bit CRC error-detecting sequence,
- "data compression" to reduce the number of bits needed to send each character,
- adaptive choice of data rate (100 or 200 bit/s) in response to special ACK or NACK frames, and
- combining (addition) of quantized and stored signal levels corresponding to bits in repeated frames ("memory ARQ," a soft-decision technique that leads to more accurate demodulation).

The soft-decision approach used in PACTOR leads to a significant increase in throughput at low signal-to-noise ratios. In addition to its higher throughput, the new system allows transmission of ASCII characters and binary data, and automatic CW identification in accordance with regulations.

10. The future

The CLOVER and PACTOR developments suggest the direction HF modems for amateur use will take in future: Modulation types, information rates and error control techniques will be programmable by the user and therefore *adaptable to channel conditions.* (Some commercial modems are already being built with this capability.) With the falling prices of signal-processing hardware in general, the prices of programmable equalizers may fall enough for them to be used by amateurs. We can expect manufacturers to produce chip-mounted software to carry out automatic link establishment according to established standards or using ALE protocols that can be adjusted by *amateur and other users themselves* (see the notes at the end of this part for an example of such a protocol).

The falling prices of powerful DSP chips—for example, those of the Texas Instruments TMS320 family—will soon make it possible to design a complete modem in software for effective HF digital operation. (Source code for a software version of the Bell 103 FSK modem used in amateur HF packet radio—and including an equalizer—has recently been published by Texas Instruments for its TMS32C17 DSP chip; chips for ALE, FEC and buffering need only be added to produce a modem that is fully capable of performing all the functions needed for effective operation in many HF channels.

Although chip sets programmed to carry out ALE according to 141A are not yet available widely and cheaply, publication of the ALE standard should make that only a matter of time. (One modem company has promised a card-mounted 1045 ALE modem for less than $1000 by the end of 1992.) When such a modem is combined with the PC and transceiver that are already part of many commercial and amateur stations, *all the hardware necessary for the beginning of true HF networking* will exist at hundreds of stations. We should then see the rapid advance of techniques that could again make HF a credible competitor of satellites in many applications.

With the publication of Federal Standard 1045 we can also expect to see more networks with *point-to-point* links set up using the new ALE protocol. As these networks produce data (similar to that shown above) on real link quality and linking probability in various conditions, further work on the more advanced standards will lead to refinements in link quality assessment, connectivity exchange and finally, network routing techniques. Techniques that cannot be adequately studied with real networks (such as certain adaptive routing strategies, inter-networking, and so on) will continue to be analyzed by means of network simulations of the type I'll treat in the next part.

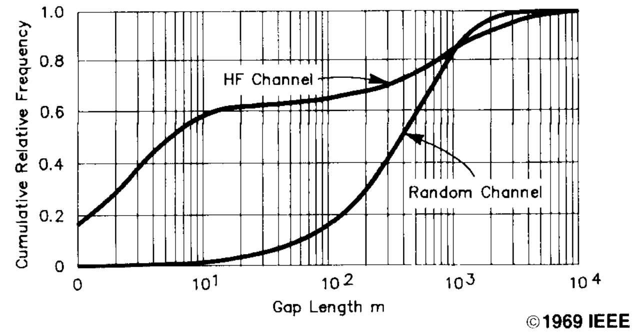
Notes to Part II

A relatively painless introduction to HF modems and their performance can be found in *HF Communications: A Systems Approach*, by N. M. Maslin (Plenum Press, New York, 1987). More details can be found in *Digital Communications* by B. Sklar (Prentice Hall, New York, 1988).

Although the human brain is a very good demodulator for low-speed digital signals, there are very few skilled Morse operators left in American governmental and commercial communications services. Although the causes of this are complicated, having to do in part with economics, they are probably also unavoidable, so that training of more skilled Morse operators can no longer be considered as a means for providing accurate, low-speed digital communications outside the amateur service.

Descriptions of the CLOVER and PACTOR modems can be found in 1990 and 1991 issues of *QEX*.

Fig 2-10 shows the cumulative probability distribution of the gaps (measured in numbers of bits) between bit errors in an actual long-distance HF channel and a simulated purely random (≈ Gaussian) channel with the same average bit error rate of about 1.8 x 10^{-3}. You can see from the figure that short gaps occur much more frequently in the HF channel than in the random one: the shortwave channel has more structure than the Gaussian channel.



©1969 IEEE

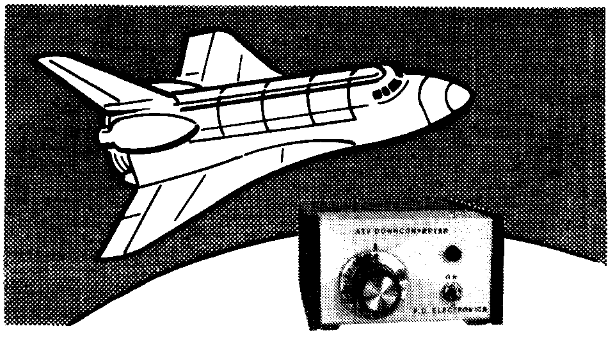
Fig 2-10 Bit Error Gap Length Distributions for Random (AWGN) and Bursty (HF) Radio Channels. (S. Tsai, IEEE Trans. on Comm. Techn., COM-17, 1969.)

The modern trend in digital signal processing for band-limited radio communications is to combine modulation and coding schemes. This is currently accomplished with so-called *trellis code modulation.* In this technique, the redundancy necessary for error control is achieved by increasing the size of the signaling alphabet by the use of multi-level or multi-phase signaling (as with QAM), which does not increase the bandwidth. The underlying idea is to increase the minimum distance (in the coding sense) between signals that are *most likely to be confused by the decoder*, but not to increase the power. The price paid for this is a very complicated decoder.

Additional FEC may be applied by a modem (or a controller) to message frame headers to give further protection to important header information.

For an example of an ALE protocol developed by amateurs see *A Selective Calling Decoder for MF/HF Radio Systems*, by Paul Newland, AD7I, in the April 1989 *QEX*.

Acknowledgment. I am grateful to my colleagues Ron Dugay, Bob Levreault, W1IMM, and Alec Osborne at the MITRE Corporation for indispensable assistance with setting up and running the ALE experiments mentioned in Section 8.1.



T/R Switching Low-Power 903 and 1296 Transverters

By Zach Lau, KH6CP/1
ARRL Laboratory Engineer

This is the latest circuitry I've used on my no-tune transverters to do the RF, IF, and dc switching. The goal was to do things as simply and cheaply as possible while maintaining reliability and performance.

The big change compared to my August 1988 *QEX* design is the decision to use the IF cable to also carry the T/R switching signal. This allows one to switch between transverters by merely switching one cable. The disadvantage is that many IF transceivers will have to be modified, though the IC-2AT works just fine for FM work. The modification consists of inserting a dc decoupling capacitor between the antenna jack and the rest of the circuitry, and using an RC low-pass filter to bring in a switched transmit voltage as shown in Fig 1. Ideally, this would be a sequenced voltage, one that comes up before RF appears and stays on until after the RF disappears, though this really isn't necessary when switching low power levels well under 1 watt.

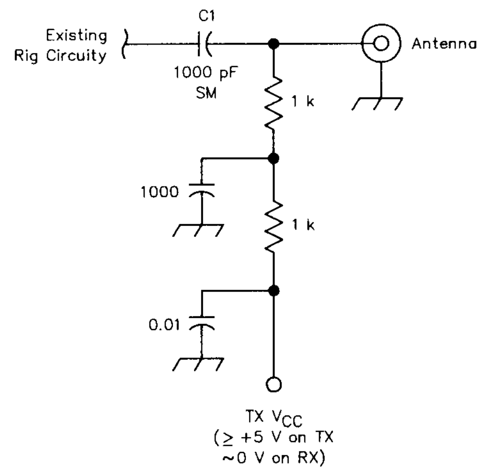


Fig 1—Simple circuit to put a switching voltage on the IF cable.
C1—Use a high quality capacitor that will handle the transmit power and not degrade your receive sensitivity with excessive loss.

Another change is the use of 1N4148 diodes for the IF switching. Since the losses aren't a problem, ordinary 1N4148s can be used with a minimum of performance degradation. With 10 volts at the 470-ohm bias resistor, an input intercept in excess of +56 dBm was measured. Since I don't know of a cheap preamp with a better dynamic range, no effort was made to eliminate the effect of the HP8640's leveling circuit, which has been known to degrade IMD measurements made by the unwary. Even with just 5 volts applied, the input intercept was still a good +47 dBm. According to the Sylvania specification sheet, 1N4148 diodes are good for 30 mA at 150 degrees C, and up to 160 mA at 25 degrees C (absolute maximum). This is better than the 1N914, which has current ratings of 10 and 75 mA, respectively. The isolation was measured at 33 dB between the transmit and receive circuits. Given the measurements, I don't see a technical reason for using more costly and harder to obtain PIN diodes in this application. The advantage of PIN diodes, the lower loss, becomes meaningless when terminated with a resistive pad!

The output RF switching is done with Omron high-frequency relays designed for 900 MHz. As of early 1992, they are only $6 each. Unfortunately, I don't really know how much power you can hot switch with these relays. Hot switching can be avoided with a sequencer, but a typical design is more complicated than all the circuitry shown here! I generally use a sequencer when running more than a few hundred milliwatts.

The dc switching is done with P-channel HexFETs. It might be cheaper to use a relay, but I think the solid state circuit is more reliable. For troubleshooting, I've gotten away with momentarily hooking up the input to +12 volts, and having the charge on the capacitors hold the transverter in transmit for long periods of time. The bypass capacitance seems to be enough to prevent static damage. The FETs have an on-resistance rather than a semiconductor junction voltage drop, so the voltage drop is current dependent. However, a voltage drop of a fraction of an ohm times a fraction of an amp is small enough in many cases to be ignored.

Construction

While a compact transverter can be built by soldering 1-inch brass walls around the circuit board, you first want to prepare the board so you can hook up RF con-

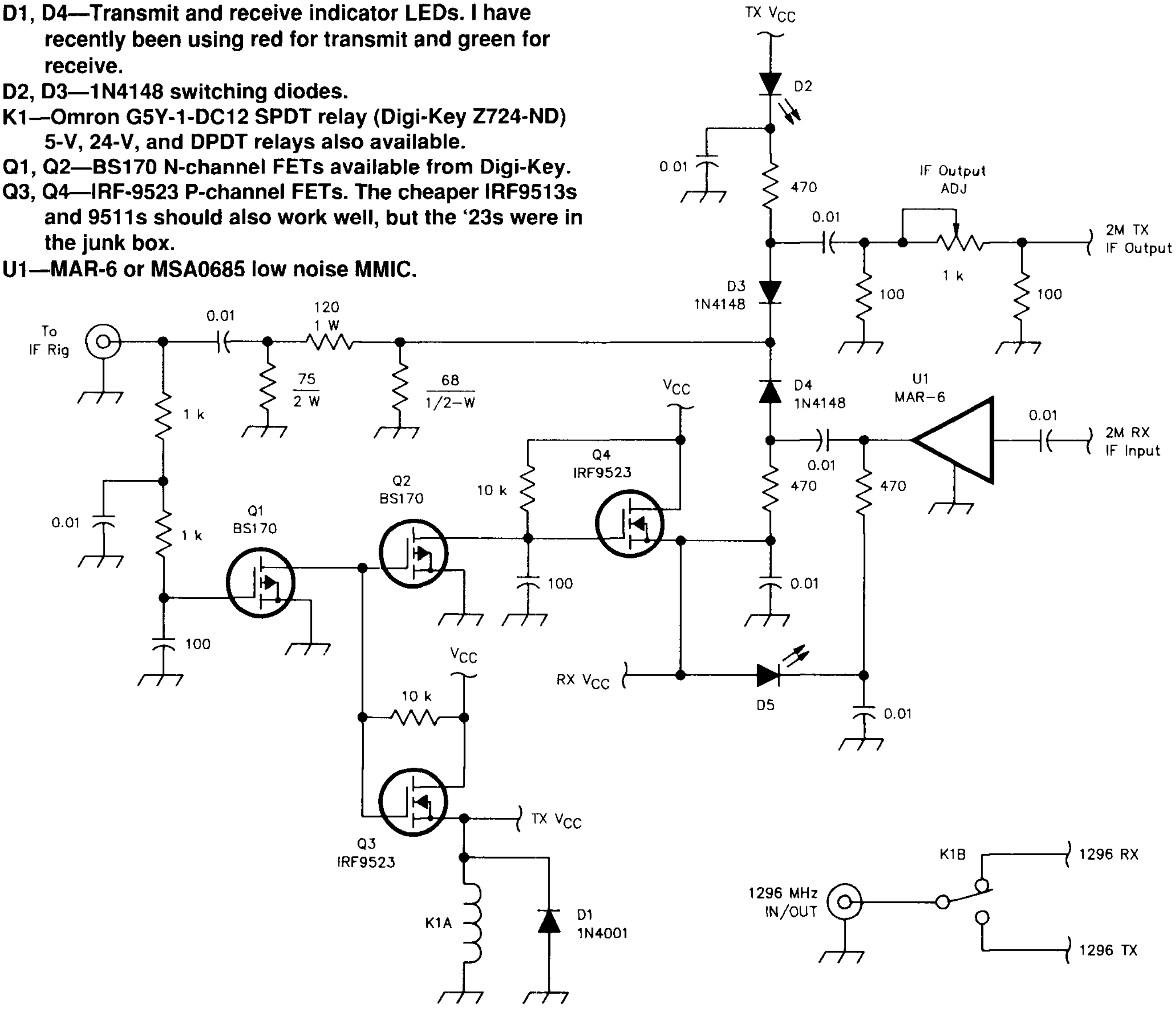


Fig 2—T/R switching circuitry for transverters.

D1, D4—Transmit and receive indicator LEDs. I have recently been using red for transmit and green for receive.

D2, D3—1N4148 switching diodes.

K1—Omron G5Y-1-DC12 SPDT relay (Digi-Key Z724-ND) 5-V, 24-V, and DPDT relays also available.

Q1, Q2—BS170 N-channel FETs available from Digi-Key.

Q3, Q4—IRF-9523 P-channel FETs. The cheaper IRF9513s and 9511s should also work well, but the '23s were in the junk box.

U1—MAR-6 or MSA0685 low noise MMIC.

nections from the back of the board. Not only do you have to drill holes and clear the ground foil, but you also need to trim at least 1/16th of an inch from the ends of the traces to prevent them from shorting to the walls. Actually, 1-inch walls are a little tight on the 1296 board if you decide to shield the LO board, as mine has threads cut into the 1296 board to accommodate the mounting screws. While double-sided glass epoxy board makes a better thermal shield than brass, using it unwisely may result in difficulties grounding the cover.

An IF BNC connector, miniature molex connector, and indicator LEDs were mounted on the front panel, while a BNC was mounted on the back panel for the antenna connection. The circuit was tacked to the back of the no-tune board. To reliably mount the power FETs, standard TO-220 insulating hardware was used to attach them directly to the 903 board. For the 1296 board, I mounted the transistors on a piece of brass bent into a shallow "U" and soldered the brass to the no-tune board. I tapped the brass, though you might want to solder regular nuts to the brass if you have a habit of stripping threads. For really high-class threaded holes, you can buy PEM nuts from Small Parts, Inc. These nifty devices can be pressed but not hammered into thin sheet metal. Teflon coax was used to connect the relays, making it easier to keep the lead lengths short without unwanted shorts.

Conducted by: Stan Horzepa, WA1LOU
75 Kreger Drive
Wolcott, CT 06716-2755

ARRL COMPUTER NETWORKING CONFERENCE NOVEMBER 7

The 11th ARRL Amateur Radio Computer Networking Conference, hosted by the Radio Amateur Telecommunications Society, will be held at Fairleigh Dickinson University in Teaneck, New Jersey, on November 7.

Call for Papers

The deadline for receipt of camera-ready papers for the conference is September 21. Those planning to submit papers should contact Lori Weinberg at ARRL, 225 Main Street, Newington, CT 06111, telephone 203-666-1541, fax 203-665-7531, for paper guidelines and/or an author's package. The ARRL needs to know the topic of your paper and whether or not you plan on attending the conference.

Topics will include, but are not limited to, digital signal processing, digital speech, packet-radio satellites, packet-radio services, HF packet-radio investigations, protocols, network development, future systems, hardware and software.

AMSAT-UK 7TH ANNUAL COLLOQUIUM STARTS JULY 30

AMSAT-UK is sponsoring a symposium focused on all OSCAR satellites and digital communications. Question and answer sessions with various amateur satellite experts are planned. As in years past, the University of Surrey will be the meeting place for this colloquium. The following is the preliminary lineup of programs planned for the weekend of July 30-August 2:

July 30: International Satellite Day
July 31: Satellite Sessions
August 1: Satellite Sessions
August 2: Satellite Sessions

Overseas members can arrive on Wednesday and leave on Monday morning if they wish. The POSH Restaurant and Bar will be used for the Saturday night shindig. Ron Broadbent wonders if the "Junk Sale" Saturday night is the right thing to do. Send your wish list to Ron at his address below.

For more details on the colloquium, contact:
Ron Broadbent, G3AAJ
94 Herongate Road
Wanstead Park
London E12 5EQ England
telephone: 081-989-6741 (09:30-18:30 UTC daily)
fax: 081-989-3430 (24 Hours)

—from AMSAT

NOS INTRO

Since releasing *NOSview* (the on-line documentation package for *NOS*) last September, a number of people have asked about the availability of the new book *NOSintro* which was mentioned in *NOSview*. The status is that it is almost ready for publication. Unfortunately, it took much longer to write than originally planned, for two reasons:

1. I have had virtually no free time over the past few months to put in the required effort, mainly because of one significant distraction: earning my living. I guess I have to be thankful for that!
2. I'll be honest. I underestimated the large amount of time needed to dot the i's and cross the t's of the final manuscript.

The good news is that we are now on the home straight and the presses should be rolling soon. I hope you will find the wait was worthwhile.

To whet your appetite, an extract from the "Big Picture" chapter of *NOSintro* follows.

NOS*view*

A quick word on *NOSview*. The current release is still the original [137]. I had planned an update at the beginning of this year, but the small number of additions and changes suggested by readers did not make it worthwhile. The plan now is to release the next version of *NOSview* at the same time as the book *NOSintro*.

An apology, by the way, to those of you who had problems installing *NOSview*. I checked the batch files thoroughly before releasing the package, using my usual DOS command interpreter *4DOS*. It turns out that *4DOS* is quite happy with symbol references like %NOSROOT, that is, with a leading % symbol and no trailing %, but the regular *COMMAND.COM* needs both a leading and trailing %, that is, %NOSROOT%. Seems there are people out there still using *COMMAND.COM*. I will do better next time (and I will even use ZIP subdirectories to set up the file tree automatically)!

Good luck with *NOS* and TCP/IP.

73 de Ian Wade, G3NRW @ GB7BIL

NOS: The Big Picture

(This is an extract from Ian's new book NOSintro *to be published later this year.)*

NOS is a multitasking operating system that provides an extremely flexible and powerful set of communications services for use on packet-radio networks, telephone lines and local area networks. *NOS* supports most of the commonly used Internet protocols such as TCP, IP, TELNET, FTP, SMTP and so on, plus the packet-radio protocols AX.25, NET/ROM and PBBS mail.

With *NOS* you can communicate with virtually any kind of computer. An Amstrad can talk to an Apple, an Amiga can talk to an IBM mainframe, a laptop PC can talk to a Cray, and so on. What's more, you can send electronic mail via worldwide networks and you can even log into remote systems as if you were directly connected to them.

NOS supports the TCP/IP AMPRnet (AMateur Packet Radio network), which rides on the back of the Internet protocols. These protocols are operating system independent. This means that you can run *NOS* on a PC running DOS or *UNIX/XENIX*, or a DEC VAX running *VMS*, or a Sun SPARC workstation running *SunOS* (or on several other platforms listed below). You can send binary or ASCII files between them, handle mail and set up gateways to link different types of network.

Probably the most important aspect of *NOS* is that all of these protocols and services conform to internationally agreed standards and are available in one form or another on virtually every micro, mini and mainframe system in use today. This means that you are not locked into special software (such as *YAPP* or *7PLUS*) that nobody outside the amateur world understands and you can communicate with almost any type of computer in the world in exactly the same way.

There's more. As well as supporting TCP/IP and AX.25, *NOS* also understands NET/ROM; you can even set up your own NET/ROM node if you want to. Why support NET/ROM? Well, in the ideal world, all *NOS* systems would talk TCP/IP directly to each other and would handle node-to-node routing at the IP level. Unfortunately, we have not yet reached this ideal state and, in most cases, we have to rely on existing networks to carry our TCP/IP traffic instead. The most widespread packet-radio network which already exists is NET/ROM, so that is why *NOS* supports it.

Thus, when you monitor TCP/IP traffic, you may see AX.25 frames which contain NET/ROM packets which contain IP packets which contain TCP packets. (Read that sentence again!) Sounds complicated, but once you have read this book you will see that it is really quite straightforward to set up, provided you keep a clear head and understand the functions of the different network layers.

And there's even more. *NOS* also supports PBBS forwarding and reverse forwarding, allowing us to communicate with the established PBBS mail network. *NOS* stores the PBBS mail files in the same directories as TCP/IP mail files and you can read and send mail in either format. Thus, we have the best of both worlds; we can choose to send our mail either via the AMPR network or via the AX.25 PBBS network and we can read mail from both of those networks as well.

The Internet Protocols

The two networking protocols at the heart of *NOS* are the Transmission Control Protocol (TCP) and the Internet Protocol (IP). These protocols were developed under the aegis of the Defense Advanced Research Projects Agency (DARPA) in the United States and have been in general use in data networks throughout the world for many years.

The raw TCP and IP protocols are not of very much interest by themselves, at least not while you are still learning how to set up NOS. What is much more important is the set of network services using TCP/IP, which you will use to transfer files, send mail and so on.

The five main classes of network services which you will use are:

- Chat
- Remote Login
- File Transfer
- Mailers
- Network Support

Let's take a brief look at these.

Chat

The chat service lets you do just that, using the command "ttylink" (or "chat" in some versions of *NOS*). Thus if you want to chat to NS9KEN, you give the command "ttylink ns9ken," and once you are connected, you can converse in exactly the same way as in plain-vanilla AX.25. *NOS* puts keystrokes in a buffer as you type and then transmits the buffer when you hit <CR>.

Remote Login

There are two different remote login services provided in *NOS*. The one you are most likely to use is TELNET. When you give a command such as "telnet ns9ken," you will normally be connected to his NOS BBS, where you can read and send mail and use various network gateways if you have permission.

The alternative login service is "rlogin." This command lets you perform a login to a remote computer which supports the RLOGIN protocol.

File Transfer

The command for file transfer is "ftp." To transfer files between your system and NS9KEN's system, you give the command "ftp ns9ken" and when you are connected, you can use the "get" command to fetch a file from NS9KEN (for example, "get yourfile.txt") or use "put" to send a file to NS9KEN (for example, "put myfile.txt").

You can transfer ASCII or binary files simply by giving the "ascii" or "binary" command before starting the transfer. You don't need to worry about lost or duplicate packets; FTP takes care of error detection and correction, so when the transfer is done, you are sure that it was successful.

Mailers

The basic function of a mailer program is to let you compose messages and bulletins ready for forwarding and to read incoming mail. There are no less than four mailers which you are likely to come across in *NOS* systems:

- BM
- ELM
- PCElm
- NOS BBS

The first three mailers are not actually part of *NOS*, but are separate programs which you can call from *NOS* when you want to access your mailbox. Alternatively you can use them completely independently of *NOS*, starting them from the DOS command line. The fourth mailer, NOS BBS, is built into *NOS* and has several extra features in addition to handling mail.

Why so many mailers? It is really a matter of history. In early versions of *NOS*, there was no built-in mailer and *BM* (for *Bdales Messy Mailer* from N3EUA) was provided instead. This had very basic functionality and was cumbersome to use, but it served its purpose at the time.

Next in line came *ELM*, which provides a much nicer full-screen menu environment. With *ELM* you can compose mail using your favorite text editor, include files in your messages, set up mailing lists and so on. Many people use this mailer today.

PCElm is a more recent mailer which looks and works very much like *ELM*, but is, in fact, unrelated. As well as providing all the facilities of *ELM, PCElm* also has a built-in text editor and lets you set up screen colors, define message file name extensions and delimiters, filter out unwanted headers and so on.

The built-in NOS BBS contains a simple mailer which works in a very similar way to the familiar AX.25 PBBS. You give commands like "L" to list mail, "SP" or "SB" to send it, "R" to read it and so forth. However, the NOS BBS also provides a set of gateway commands which let users break out into the NET/ROM network or telnet into another NOS BBS or even take over control of your station. You will be glad to hear they can not do any of these things unless you give them permission first!

Use of the NOS BBS is not restricted to TCP/IP users. An ordinary AX.25 user can connect to your NOS BBS, read and send mail like a TCP/IP user and can use the gateway commands as well if they have permission. In other words, this gives an ordinary AX.25 user the capability of accessing the NET/ROM network and AMPRnet if they want.

So, which of these four mailers do you use? If you are logging into someone else's system, you have no choice: The built-in NOS BBS on that system is the only mailer you can access. On your own system you can use the built-in NOS BBS if you want, but you'll probably prefer to use *PCElm* (or perhaps *ELM)* as it has a much nicer user interface.

Mail Forwarding

The main function of the mailers is to let you read and write mail. To send and receive this mail, *NOS* provides three mail forwarding services:

- Simple Mail Transfer Protocol (SMTP)
- Post Office Protocol (POP)
- AX.25 PBBS Forwarding

SMTP handles the sending and receiving of mail via AMPRnet and is the default method of handling mail. Using SMTP you can transfer mail between any computers which understand it, that is, virtually any kind of machine in the world.

POP is the reverse forwarding protocol that works with SMTP. With POP you can nominate another machine as your "Post Office" and when you run POP, your own machine will automatically login to the Post Office and retrieve any mail waiting for you.

In addition to running SMTP and POP, you can also run AX.25 PBBS forwarding and reverse forwarding. This is fully compatible with the PBBS network, so if you do not have access to a local *NOS* system which can forward your mail using SMTP, you can still communicate with the outside world via the PBBS network.

Network Connectivity Services

NOS provides a number of network support services which let you check the availability of other stations on the network. These services include:

- ping
- hop

The "ping" command, known officially in the trade as the Packet Internet Groper (...amazing but true), is useful when you are not sure if a local station is responding to your traffic. Whenever you want to check if a local station is active, you ping it, for example, "ping ns9ken." If NS9KEN is running *NOS,* it will respond to your ping and you will see on the screen a number representing the round-trip time in milliseconds for your ping packets. If you get no response or if the round-trip time is unexpectedly long, you know that something is wrong.

The "hop" commands let you check the availability of routes to a particular station. For example, to find out which gateways your packets pass through to reach NS9LIZ, you would give the command "hop check ns9liz." This is very useful to verify that a route exists to the target station (and can sometimes show up some bizarre routes that you never knew existed).

Network Addressing

Every *NOS* station has an IP address, a unique 32-bit number which is usually expressed as four decimal numbers separated by dots (the so-called "dotted decimal" notation). For example, NS9BOB's IP address is 44.131.49.1. The first byte is always 44, which represents the AMPR network. The second byte (131) represents a country (or a state in the United States). The third and fourth bytes are an address within that country or state. Typically, the third byte will represent a region or area and the last byte will be a station number in that region.

Incidentally, you may see some documentation which shows IP addresses enclosed in square brackets, for example, [44.131.49.2]. This convention is a relic of early *NOS* systems, and is not used today.

Each country or state where there is AMPRnet activity has a local IP address coordinator who allocates IP addresses on request. If your country does not yet have a coordinator, you should contact the international coordinator in the United States instead (but be prepared, he will probably nominate you as your countryUs coordinator).

From time to time the coordinators issue a full list of IP addresses in their area. When you set up your *NOS* station, you will use this list to create the file DOMAIN.TXT, which *NOS* uses whenever you make a network connection. (Strictly speaking, you don't really need a DOMAIN.TXT file, you could use IP addresses instead of symbolic hostnames, for example, you could give the command "ping 44.131.49.2" instead of "ping ns9ken," but obviously it is more meaningful to use names rather than addresses.)

Keeping DOMAIN.TXT up to date is clearly a problem. One way around this is to nominate a local station as a domain server, which keeps a master copy of the file and makes it available to other users. (This is somewhat similar to a PBBS White Pages server, which keeps a record of AX.25 stations and their local mailboxes.)

If you then set up *NOS* to use the domain server and attempt to make a network connection, *NOS* will first look in your own DOMAIN.TXT file for the hostname you have given. If it can't find the hostname there it then automatically make a request to the domain server machine for the IP address of the station you are trying to contact.

Address Resolution Protocol

When setting up a network connection, *NOS* needs to know not only the IP address, but also the link address of the station you wish to talk to. If you are using a radio link, the link address is the other station's call sign, for example, NS9KEN-5. If you are on Ethernet, the link address is the 48-bit hardware address of the Ethernet adapter card in the other station's PC, for example, 11:22:33:44:55:66).

To set up the table of link addresses, *NOS* provides the Address Resolution Protocol ("arp") set of commands. For example, to communicate with NS9KEN, you could give the command "arp add ax25 ns9ken NS9KEN-5," thus forming an association between the IP hostname (ns9ken) and the link address (NS9KEN-5).

Routing

Routing controls how packets get to their destination. *NOS* supports no less than three completely independent levels of packet routing:

- AX.25 routing
- NET/ROM routing
- IP routing

You are already familiar with AX.25 routing, particularly when it's referred to by its more usual name: digipeating. *NOS* has a set of "ax25 route" commands which let you set up digipeater paths to nominated destinations. For example, to route packets via digipeater AX9DIG-0 to reach NS9LIZ-5, you simply say "ax25 route add NS9LIZ-5 AX9DIG-0."

Similarly, there is a set of "netrom route" commands, for setting up NET/ROM routes and aliases which ordinary NET/ROM nodes understand.

IP routing is basically an extension of the NET/ROM routing idea, specifically for forwarding IP packets onwards to their final destination. *NOS* has a set of "route" commands for setting up and maintaining the IP routing table.

Routing Updates

In the amateur packet network, nothing lasts forever or even for a lot less time than ever! Routes between nodes are continually changing as stations come and go, frequencies change and so on. This means that for there to be any realistic chance of communicating with other users on the network, your station has to be kept up-to-date with the current routing situation.

NOS achieves this in two ways. First, it listens to user traffic on the frequency and when it hears stations, it dynamically updates the appropriate routing tables. These updates remain in memory for a finite period (usually measured in minutes or tens of minutes) and, if a station is not heard again, the routing information for that station eventually disappears.

The second way that *NOS* keeps up-to-date is by routing broadcasts. *NOS* regularly sends broadcasts of the NET/ROM routing table in the same way as a native NET/ROM node and also sends IP routing table broadcasts at regular intervals.

For IP routing broadcasts, *NOS* supports two protocols: Routing Internet Protocol (RIP) and Radio Shortest Path First (RSPF) protocol. RIP is the protocol to use if your station is part of a classic and fairly stable Internet environment (for example, on Ethernet), whereas RSPF works better in a dynamic radio environment.

Wormhole Routing

Another method of packet routing supported by *NOS* is the wormhole, which provides plain-vanilla AX.25 connectivity over a TCP/IP link. This is useful where you are linking two AX.25 stations via an Ethernet or telephone connection. In effect, the *NOS* wormhole acts like a (rather complicated) digipeater.

Interface Support

NOS is noted for its very wide range of supported interfaces (although not every version of *NOS* will support all of them). These include:

- The serial ports (COM1 - COM4) for TNCs or modems
- Modem control
- Ethernet adapters
- Clarkson drivers
- DRSI PC*PA 8530 card
- HAPN 8273 adapter
- High speed DRSI/HAPN driver
- Eagle 8530 card
- NET/ROM control
- Single- and multi-port KISS TNCs
- PacComm PC100

In other words, *NOS* will talk to virtually any TNC or modem or any of the well-known network adapters. The Clarkson drivers are freely available as public domain software and support all of the generally available Ethernet and Token Ring LAN adapters.

NOS talks a number of low-level protocols via these interfaces:

- KISS for TNC control
- SLIP and PPP for serial point-to-point telephone links
- NRS for NET/ROM control
- Ethernet and ARCnet for Ethernet adapters

The NOS Session Manager

Because *NOS* is a multitasking system, you can run many sessions in parallel. Hence, it is possible, for example, to telnet to NS9KEN, do file transfers with NS9LIZ, access your own mailbox, ping NS9BOB and chat with AX.25 station AX9AAA, all at the same time. (In principle, you can run many more sessions as well, but you need a Jekyll and Hyde personality to handle it all!)

The NOS Session Manager maintains a virtual screen and keyboard for each session and you can hot-key from session to session at will. You can find the status of any session on demand and you can trace all the traffic flowing through your station, right down to the hexadecimal byte level if you want to. You can also record any session on disk for later use. Furthermore, you can allow other stations to drive your NOS Session Manager remotely if, for example, your station is on a remote hilltop site.

That's NOS

By now, it will be clear that *NOS* is a very complex package with many advanced features that make it usable in a wide variety of environments. To the beginner, some of the features in *NOS* may appear to be daunting, but fortunately it is not necessary to understand everything before you can use it. Just as you first use a TNC, you can get away with using default setup parameters; performance may not be optimum, but it will at least work. Then as you gain experience, you can dig deeper into the software and start to experiment with different configurations.

Driving *NOS* is a bit like driving a car. Think of the network protocols (TCP, IP, SMTP and so on) as the engine and the network services (like telnet and ftp) as the brakes and steering. To drive a car, you have to know about the brakes and steering, but learning about engine valve timing and compression ratios can wait until later. To drive *NOS,* all you really need to know at first is how to set up the TNC, configure the address and routing tables and use the basic network services to transfer files and handle mail. Fine-tuning of the network protocol parameters can wait until much later.

NOS *Software Availability*

NOS is now available on many platforms, including the Atari, Amiga, Archimedes, *PC-DOS,* Macintosh, DEC VAX/VMS and Ultrix, SCO *UNIX,* SCO *XENIX,* Interactive *UNIX* and Sun *SunOS.* This book describes the DOS version 1.8b from Gerard van der Grinten, PA0GRI, dated 29 December 1991.

CORRECTION: NEW PACKET-RADIO SATELLITE SURVEY

The April Gateway *Survey for New Packet-Radio Satellite* asked readers to complete a survey concerning a new packet-radio satellite proposed by the Naval Postgraduate School space systems department. The completed survey was to be sent to the school's Internet address. Problem is that the Internet address was truncated. The complete address is: 7996p@cc.nps.navy.mil

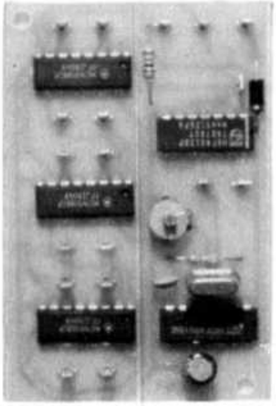
DISCLAIMER

The descriptions of hardware and software contained in this column are merely reports of what is being offered in the Amateur Radio marketplace. Your Gateway contributing editor, *QEX* and the ARRL in no way warrant the operability, functionality, suitability, or even the existence of what is being offered.

GATEWAY CONTRIBUTIONS

Submissions for publication in Gateway are welcome. You may submit material via the US mail to 75 Kreger Dr, Wolcott, CT 06716-2755, or electronically, via CompuServe to user ID 70645,247, or via Internet to horzepa@evax.gdc.com. Via telephone, your editor can be reached on evenings and weekends at 203-879-1348 and he can switch a modem on line to receive text at 300, 1200 or 2400 bit/s. (Personal messages may be sent to your Gateway editor via packet radio to WA1LOU@N4GAA or IP address 44.88.0.14.)

The deadline for each installment of Gateway is the first day of the month preceding the issue date of *QEX.*



Bits

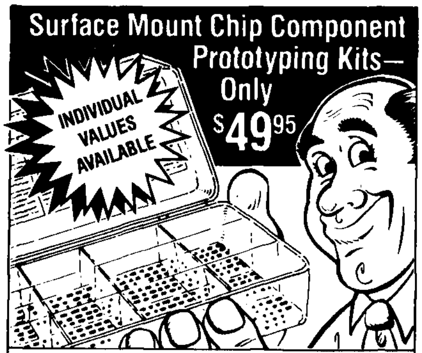
New Helix Antenna Design

A two-turn axial mode helical antenna design is described in the IEEE Transactions on Antennas and Propagation for June 1991. A paper, entitled "Extremely Low-Profile Helix Radiating a Circularly Polarized Wave," by H. Nakano, et al, describes a 9-dB gain helix design with an axial length of only 0.19λ(ie, 1.3 feet at 145 MHz). This one wavelength circumference helix is only 0.14 λ long, corresponding to a 4° pitch angle, and has 0.05 λ ground plane spacing providing a very compact design.

This antenna should prove useful for portable operation or by amateurs who have space constraints, especially on 2 meters where conventional designs are 6 feet or longer.

This article can be obtained at your local library or from IEEE, 445 Hoes Lane, PO Box 1331, Piscataway, NJ 08855-1331. —*Bill Shanney, KJ6GR, 19313 Tomlee Avenue, Torrance, CA 90503.*

(Continued on page 24)



CC-1 Capacitor Kit contains 365 pieces, 5 ea. of every 10% value from 1pf to .33μf. CR-1 Resistor Kit contains 1540 pieces; 10 ea. of every 5% value from 10Ω to 10 megΩ. Sizes are 0805 and 1206. Each kit is ONLY $49.95 and available for Immediate One Day Delivery!

Order by toll-free phone, FAX, or mail. We accept VISA, MC, COD, or Pre-paid orders. Company P.O.'s accepted with approved credit. Call for free detailed brochure.

COMMUNICATIONS SPECIALISTS, INC.
426 West Taft Ave. • Orange, CA 92665-4296
Local (714) 998-3021 • FAX (714) 974-3420
Entire USA 1-800-854-0547

RFI Problems?

***Fiber Optics* for computer networking. Great for ...**
- **CT multiop networking**
- **PC to TNC connections**
- **PC to external modems**
- **PC to PC networking**

A Fiber Optic Solution
- RFI/EMI immunity
- Lightening protection
- Greater transmission distance
- Smaller, lighter transmission line
- No ground loop problems or voltage spikes

Works with your computer serial port - fully transparent operation - nothing else needed! Ideal for use between your computer and TNC or modem. Runs up to 200 feet. Typical configuration: 20' link, 2 multiplexors from $136.00 - 12 VDC req. - Call or FAX for prices - 914-687-9700

King Enterprises - PO Box 11, Cottekill, NY 12419

Serious About AMTOR?

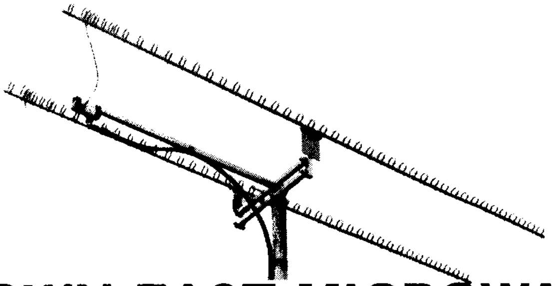
Your IBM-PC or compatible, our **BMK-MULTY** software and any ordinary RTTY terminal unit such as CP-1, CP-100, ST-5, ST-6, ST-6000 etc. will give you unequaled performance. It's AMTOR done right!

- Uses the 16-bit power and large memory capacity of your PC to directly process data signals. Avoids TNC limitations by eliminating the TNC.
- Advanced AMTOR algorithm syncs extremely rapidly.
- Automatic selection of Mode-L, FEC or ARQ-QSO.
- Unique multi-sampling RTTY algorithm.

Detailed literature upon request. Matching T.U. available. Base version with AMTOR, RTTY, CW and Audio Spectrum Analyzer $95. Extended version also includes HF WEFAX and SSTV reception $125. VISA/MasterCard accepted.

Amateur callsign required with order. State 3½ or 5¼ inch disk preference.

Authorized U.S. Distributor:
Schnedler Systems AC4IW
25 Eastwood Rd. • P.O. Box 5964 • Asheville, NC 28813
(704) 274-4646



(Bits Continued from page 23)

Micropower DC-DC Converter

Linear Technology's LT1073 is a versatile micro-power dc-dc converter. The device requires only three external components to deliver a fixed output of 5 or 12 volts. The very low minimum supply voltage of 1.0 V allows the use of the LT1073 in applications where the primary power source is a single cell. An on-chip auxiliary gain block can function as a low-battery detector or linear post-regulator.

Features include: operates at supply voltages from 1 to 30 V; consumes only 95 µA supply current; works in step-up or step-down mode; only 3 external off-the-shelf components required; low-battery detector comparator on-chip; user-adjustable current limit; internal 1-A power switch; fixed or adjustable output voltage versions; and space-saving 8-pin MiniDIP or SO8 package.

Applications: pagers; cameras; single-cell to 5-V converters; battery backup supplies; laptop and palmtop computers, cellular telephones' portable instruments; 4-20 mA loop powered instruments; and handheld inventory computers.

For more information on the LT1073 micropower dc-dc converter contact Linear Technology Corporation, 1630 McCarthy Blvd, Milpitas, CA 95035, tel 408 432-1900, fax 408 434-0507.

"STEALTH" Antenna

The Forbes Group has introduced a new 2-m antenna aimed at hams (and scanner users) who have restrictions on outside antennas. The antenna mounts directly over the existing roof vent pipe, thereby exempting it from covenants and ordinances. The antenna installs in less than ten minutes, and sells for $39.95. For more information, contact The Forbes Group, PO Box 445, Rocklin, CA 95677, or phone 916 624-7069.