$5

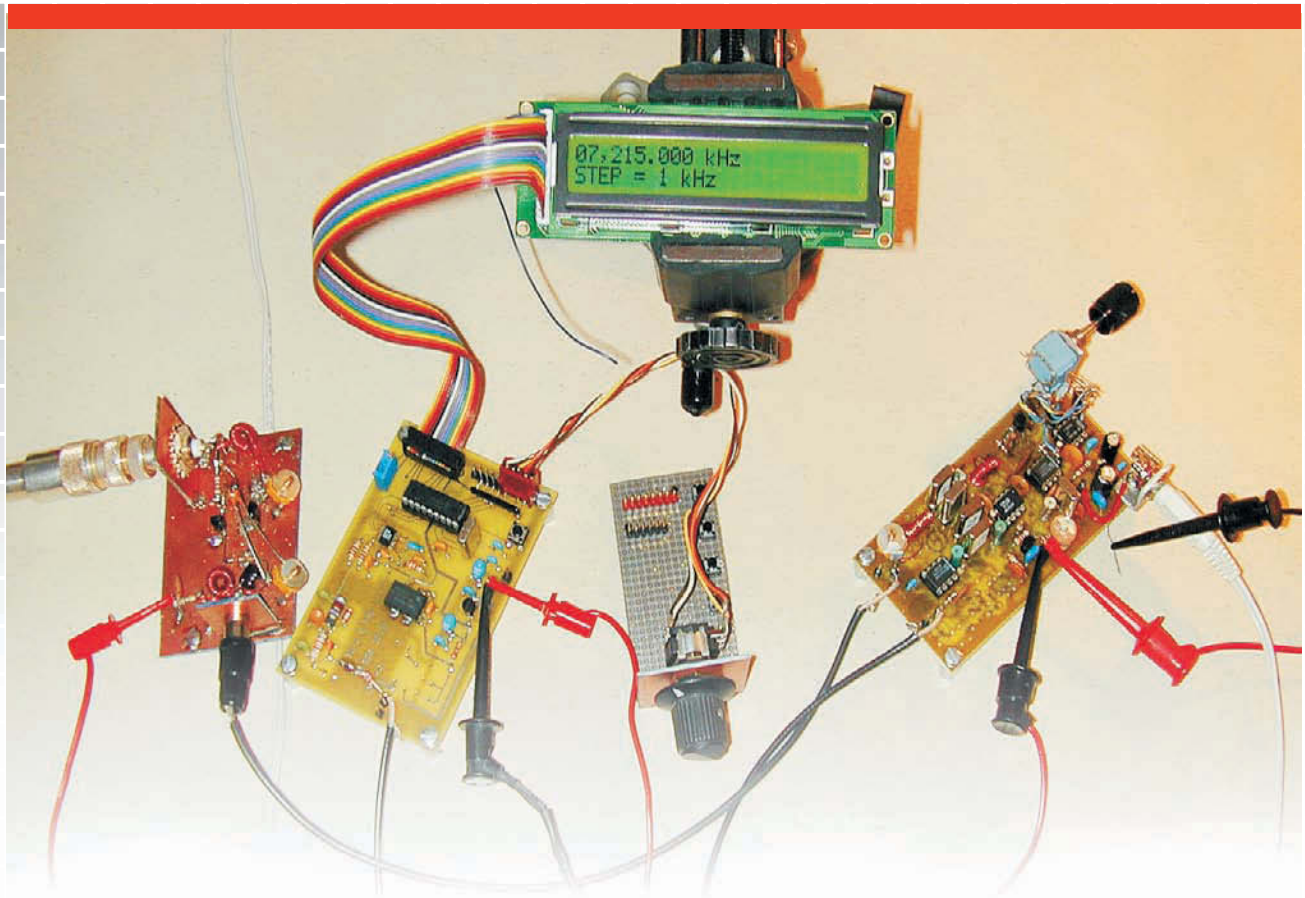
QEX

Forum for Communications Experimenters

INCLUDING: COMMUNICATIONS QUARTERLY

September/October 2005

Issue No. 232



W6IBC Presents Low Power DDS with the AD9834 and PIC

ARRL The national association for AMATEUR RADIO

225 Main Street
Newington, CT USA 06111-1494

Continuing Education

Register Online! www.arrl.org/cce

There's no better time to improve your skills. Online Classes are Available Now through the ARRL Certification and Continuing Education Program. Complete 100% of your training via the Internet:

- **Self-paced (asynchronous) format**—you attend class when and where you want.
- **High quality web experience** enhanced with graphics, audio, video, hyper-linking and interactive modules.
- **Online Mentoring**. Individually assigned instructors help advance each student toward successfully completing the course material.
- **Pre-register Now!** Classes open regularly.

Available Courses

Antenna Design and Construction…EC-009

Students become familiar with antenna design theory and experience hands-on construction techniques. The course includes several optional antenna construction projects for HF, VHF, and UHF.
Member: $65 / Non-member: $95

Antenna Modeling…EC-004

This course is an excellent way to learn the ins and outs and the nitty-gritty details of modeling antennas. In the last decade the science of modeling antennas using computer software has advanced by huge leaps and bounds. While some absolutely unique, brand-new antenna designs have resulted from computer studies, the real progress has been in our understanding of how even common, ordinary antennas work.
Member: $85 / Non-member: $115

HF Digital Communications…EC-005

Understanding HF digital Amateur Radio communications and developing awareness and stronger skills for many HF digital modes. **Member: $65 / Non-member: $95**

Level 1 Amateur Radio Emergency Communications…EC-001

Introduction to Amateur Radio Emergency Communications. A basic course to raise awareness and provide additional knowledge and tools for any emergency communications volunteer.
Member: $45 / Non-member: $75

Level 2 Amateur Radio Emergency Communications…EC-002

Intermediate Amateur Radio Emergency Communications. A more in-depth study into amateur radio emergency communications to enhance the skills and knowledge received from previous experience. Requires prior completion of EC-001.
Member: $45 / Non-member: $75

Level 3 Amateur Radio Emergency Communications…EC-003

Advanced Amateur Radio Emergency Communications. Bridging the gap between basic participation and leadership. Requires prior completion of EC-001 and EC-002.
Member: $45 / Non-member: $75

Radio Frequency Propagation…EC-011

Explore the science of RF propagation, including the properties of electromagnetic waves, the atmosphere and the ionosphere, the sun and sunspots, ground waves and sky waves, and various propagation modes—including aurora and meteor scatter.
Member: $65 / Non-member: $95

Radio Frequency Interference…EC-006

Learn to identify sources and victims of interference. Tips and suggestions for solutions and for handling those ticklish problems that crop up with difficult neighbors and other aggrieved parties. Tools to help foster ingenuity, intuition, and determination for solving interference problems.
Member: $65 / Non-member: $95

VHF/UHF—Life Beyond the Repeater…EC-008

An introduction to Internet linking, amateur satellites, direction finding, APRS, weak signals, VHF contesting, microwaves, amateur television, and high speed multimedia radio. Great for both the newly licensed and more experienced hams.
Member: $65 / Non-member: $95

Technician License Course…EC-010

The course prepares students to earn their first Amateur Radio license. There are no prerequisites. Individually assigned online mentors assist students as they advance toward successfully completing the course. Registration includes the ARRL book, *Now You're Talking!* and online graduate support.
Member: $99 / Non-member: $139

Analog Electronics…EC-012

Students learn about the use of instrumentation, Kirchoff's Laws, Diodes, Rectifier circuits, Bipolar and Field Effect Transistors, various amplifier configurations, filters, timers, Op-Amps, and voltage regulators. Most lessons include a design problem and optional construction project. (College-level course.)
Member: $65 / Non-member: $95

Digital Electronics…EC-013

Student learn about Basic Boolean, Basic Gates, Flip-Flops, Counters and Shift Registers, Latches, Buffers and Drivers, Encoders and Decoders, Parallel Interfaces, Serial Interfaces, Input Devices, Displays, Logic Families, Microprocessor Basics, Connecting to Analog Electronics, Understanding Data Sheets and Design Resources. Most lessons include a design problem and optional construction project. (College-level course.)
Member: $65 / Non-member: $95

Online courses are produced by American Radio Relay League, Inc. and are available through ARRL's partnership with the Connecticut Distance Learning Consortium (CTDLC), a nonprofit organization that specializes in developing on-line courses for Connecticut colleges and universities. Continuing Education Units (CEUs) are available for all ARRL Certification and Continuing Education courses. The ARRL Certification and Continuing Education Program is funded in part by course fees from interested hams who support public service and quality continuing education.

For further information, e-mail your questions to **cce@arrl.org**, or write to **ARRL C-CE**, **225 Main Street**, **Newington**, **CT 06111**.

ARRL *The national association for AMATEUR RADIO*
www.arrl.org/cce

QEX

INCLUDING: COMMUNICATIONS QUARTERLY

QEX (ISSN: 0886-8093) is published bimonthly in January, March, May, July, September, and November by the American Radio Relay League, 225 Main Street, Newington CT 06111-1494. Periodicals postage paid at Hartford, CT and at additional mailing offices.

POSTMASTER: Send address changes to: QEX, 225 Main St, Newington, CT 06111-1494 Issue No 232

Offices

225 Main St, Newington, CT 06111-1494 USA
Telephone: 860-594-0200
Fax: 860-594-0259 (24 hour direct line)
e-mail: **qex@arrl.org**

Subscription rate for 6 issues:

In the US: ARRL Member $24, nonmember $36;

US by First Class Mail: ARRL member $37, nonmember $49;

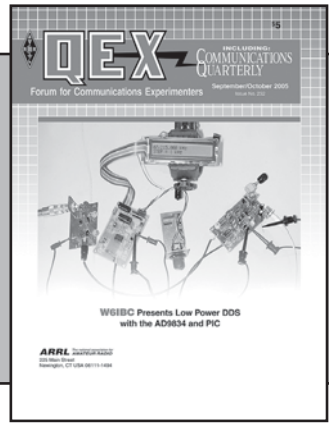
Elsewhere by Surface Mail (4-8 week delivery): ARRL member $31, nonmember $43;

Canada by Airmail: ARRL member $40, nonmember $52;

Elsewhere by Airmail: ARRL member $59, nonmember $71.

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

In order to ensure prompt delivery, we ask that you periodically check the address information on your mailing label. If you find any inaccuracies, please contact the Circulation Department immediately. Thank you for your assistance.



About the Cover

A breadboard setup of W6IBC's low-power DDS modules. The story begins on p 48.

Features

Columns

Sep/Oct 2005 *QEX* Advertising Index

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: 860-594-0200

FAX: 860-594-0259 (24-hour direct line)

Officers

President: JIM D. HAYNIE, W5JBP
3226 Newcastle Dr, Dallas, TX 75220-1640
Chief Executive Officer: DAVID SUMNER, K1ZZ

The purpose of *QEX* is to:
1) provide a medium for the exchange of ideas and information among Amateur Radio experimenters,
2) document advanced technical work in the Amateur Radio field, and
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 letters for publication in *QEX* should be marked Editor, *QEX*.

Both theoretical and practical technical articles are welcomed. Manuscripts should be submitted on IBM or Mac format 3.5-inch diskette in word-processor format, if possible. We can redraw any figures as long as their content is clear. Photos should be glossy, color or black-and-white prints of at least the size they are to appear in *QEX*. Further information for authors can be found on the Web at **www.arrl.org/qex/** or by e-mail to **qex@arrl.org**.

Any opinions expressed in *QEX* are those of the authors, not necessarily those of the Editor or the League. While we strive to ensure all material is technically correct, authors are expected to defend their own assertions. Products mentioned are included for your information only; no endorsement is implied. Readers are cautioned to verify the availability of products before sending money to vendors.

Empirical Outlook

Allocating Our Precious Resources

At its July 2005 meeting, the ARRL Board of Directors voted to adopt recommendations to regulate the use of Amateur Radio spectrum by bandwidth instead of by mode. A petition for rulemaking is expected to be filed with the FCC before long. We think this is a great idea that bodes well for current and future development of new modes and techniques. For one thing, it would remove ambiguity regarding simultaneous digital voice, data and video on HF and would allow at least 3.5 kHz for such modes. See the full story at **www.arrl.org/news/stories/2005/07/19/3/?nc=1**.

The statement in 47 CFR 97.307(a) that says "No amateur station transmission shall occupy more bandwidth than necessary for the information rate and emission type being transmitted, in accordance with good amateur practice" will be retained. Also retained is 47 CFR 97.101(a), which refers to "...good engineering and good amateur practice." One change is that maximum permitted bandwidths would be defined as necessary bandwidths rather than occupied bandwidths. It seems the underlying idea is that most hams don't have the ability to measure their occupied bandwidths, but they should be able to get a pretty good idea. It's not clear how that change will sit with the FCC but we'll see.

While a 3.5 kHz allotted bandwidth is nice for medium-speed digital modes, some would argue that we should be trying to narrow our signals instead. In the case of digital voice on HF, we currently have several systems that fit neatly within 3.5 kHz but not by much, and they do not operate well near 0 dB signal-to-noise ratios. Some of us have been discussing whether a digital voice system occupying 500 Hz or less would be feasible in an attempt to get under the noise floor. It is certainly possible because as pointed out on these pages before, a person speaking 150 words per minute with a vocabulary of 65,000 words is only generating 40 bits/s (based on transmitting only the word information).

Without additional information though, the system would sound robotic because it would transmit none of the pitch or other voice characteristics. It might be possible to add them and some error correction, though, at the cost of a little more bandwidth. Moreover, you must have a good voice-recognition engine at the transmitting end. Still, it might make an interesting experiment, resulting in some new voice-contact records. What do *you* think?

In This Issue

We must hold Rod Brink, KQ6F's article about his direct-conversion phasing rig until next time. Sometimes such things happen because we can't make all the scheduled articles fit into our 64 pages.

Joe Taylor, K1JT, explains in detail his JT65 protocol for weak-signal work. It uses a form of message "source coding" similar to the familiar Q signals of CW and to the digital voice scheme mentioned above. It also involves some other, rather sophisticated coding for synchronization and error correction. *NCJ* Editor Carl Luetzelschwab, K9LA, discusses the effects of SWR on the drain efficiency of power amplifiers. The modeled amplifier was described by its Caltech designers in our Jan/Feb 2004 issue.

Gary A. Geissinger, WAØSPM, tells how he designed a USB-controlled VFO for his Heathkit rig. The circuit is a combination of new and old technologies. Tom Cefalo Jr, W1EX, writes about a sequencer for your transverter. Andrew Roos, ZS1AN, contributes a "hybrid" antenna tuner balun that can help you maintain an efficient delivery system for your transmitter's output. John Klingelhoefer, WB4LNM, contributes the second of our pieces about HP-8640B repair. The first was by Markus Hansen, VE7CA, in the Jul/Aug 2004 issue. This time though, it's an oscillator and not an amplifier that fails. George Steber, WB9LVI, shows us his LMS impedance bridge. We have an interesting article from David Harrison, W6IBC, about his low-power DDS based on the AD9834 chip. David includes a PIC microcontroller and some code for making the whole thing go. In "Antenna Options," L.B. Cebik, W4RNL, takes us through some of the nuances of antenna modeling software.—*73, Doug Smith, KF6DX,* **kf6dx@arrl.org**. □□

The JT65 Communications Protocol

Learn all about this popular weak-signal tool.

By Joe Taylor, K1JT

JT65 is a digital protocol intended for Amateur Radio communication with extremely weak signals. It was designed to optimize Earth-Moon-Earth (EME) contacts on the VHF bands and conforms efficiently to the established standards and procedures for such QSOs. JT65 includes error-correcting features that make it very robust, even with signals much too weak to be heard. This paper summarizes the technical specifications of JT65 and presents background information on its motivation and design philosophy. In addition, it presents some details of the implementation of JT65 within a computer program called *WSJT*, together with measurements of the resulting sensitivity and error rates.

272 Hartley Ave
Princeton, NJ 08540-5656
k1jt@arrl.net

1. Introduction

Spark gave way to continuous wave some 80 years ago. More or less by default, international Morse code with on-off keying has been the mode of choice for most Amateur Radio weak-signal work ever since. Morse is convenient, versatile, and readily encoded and decoded by humans. On-off keying is trivial to implement, and the required bandwidth is small. The choice has been an easy one.

It is easy to show, however, that neither the encoding nor the modulation of CW is optimum. When every dB of signal-to-noise ratio counts, as it does in amateur meteor-scatter and EME contacts, there are very good reasons to explore other options. Personal computers equipped with sound cards provide a golden opportunity for experimenting with the wide range of possibilities. The program *WSJT*[1,2,3] ("Weak Signal communications, by K1JT") is the result of my effort to introduce much more efficient coding and modulation schemes into amateur weak-signal communications. In the program's brief existence it has already become well known to nearly all weak-signal VHF/UHF operators, and is in regular use by many of them. On the VHF bands the overwhelming majority of all meteor-scatter QSOs and perhaps half of all EME QSOs are now being made with the help of *WSJT*.

The present paper describes JT65, one of the communications protocols supported by *WSJT*. JT65 is designed explicitly for communicating with

[1]Notes appear on page 12.

extremely weak signals like those encountered on the EME path. Operational aspects of the program are described in the *WSJT User's Guide;*[4] here I will be concerned with a complete technical description of the protocol and a general description of the way it is implemented in *WSJT*.

Modern digital communication systems are based on the mathematics of information theory. This field essentially originated with two classic 1948 papers[5] in which Claude Shannon proved that information can be conveyed over a noisy channel with arbitrarily low error rate and a throughput that depends only on channel bandwidth and signal-to-noise ratio (SNR). Achieving a low error rate at very low SNR requires the mathematical encoding of user information into a form that is compact yet includes carefully structured redundancy. Compactness is necessary in order to minimize transmitter power and maximize throughput; redundancy is needed to ensure message integrity on a noisy and variable channel.

To be transmitted by radio, an encoded message must be impressed onto a carrier wave using some form of modulation. The possibilities are almost limitless: Information can be conveyed by varying the amplitude, frequency, or phase of a carrier, or any combination thereof. Commonly used digital modulation schemes include on-off keying (a limiting case of amplitude modulation), phase-shift keying and frequency shift keying. The JT65 protocol uses 65-tone frequency shift keying with constant-amplitude waveforms and no phase discontinuities. This form of modulation is much more efficient than on-off keying, especially when combined with an optimal coding scheme. In addition, it is much more tolerant of frequency instabilities than is phase-shift keying.

Section §2 of the paper begins with some background information that has helped to motivate the design philosophy of JT65, and Section §3 presents a high-level view of the overall system design. The protocol itself is defined in §4-8 and in Appendix A, while §9-12 describe the reception and decoding of a JT65 signal. The protocol specification completely defines the translation of any valid JT65 message into a waveform for transmission, and provides all information necessary for decoding a received JT65 signal. I include the essential details of how these tasks are actually carried out in *WSJT*. Different implementations of JT65, and especially the algorithms used for reception, are also feasible. I hope that this paper will motivate others to attempt this task, and that such efforts will lead to further improvements in the performance and operational convenience of this mode.

2. EME QSOs: Requirements and Procedures

Amateur Radio is a just-for-fun activity, and for many the fun has always included such goals as making contacts with all continents, all US states, and as many DXCC entities as possible. These goals are especially difficult on the EME path—and therefore, for many, all the more challenging and desirable. To make the game one that anybody can understand and play, it is necessary to agree on some basic ground rules.

When signals are reasonably strong and communication between skilled operators essentially error free, it is easy to judge whether a QSO has taken place. When a rare one shows up on the amateur HF bands, rapid-fire QSOs in the ensuing pile-up generally proceed something like the following exchange:

1. CQ HC8N
2. K1JT
3. K1JT 599
4. 599 TU
5. 73 HC8N

In this model contact K1JT never sends the call sign of the station he is working, because the situation has made this information implicit and moot. The signals may not be "S9" at either receiver, but no one really cares. After the exchange has taken place, both stations confidently enter the QSO in their logs, and they may later exchange QSL cards to confirm that the contact took place.

In the VHF/UHF world, and especially when working over the EME path, signals are often very weak and communication between even the most skilled operators is far from error free. As a result, more rigorous standards need to be adopted for what constitutes a minimum legitimate QSO. Long-established rules hold that a valid contact requires each station to copy both complete call signs, a signal report or some other piece of information, and explicit acknowledgment that all of this information has been received. These guidelines apply and work well for all types of weak-signal QSOs, whether by tropo, meteor scatter, EME, or other propagation modes, and with all types of equipment and signaling methods.

Following these guidelines closely, the minimal EME QSOs of savvy VHF operators generally proceed something like the following sequence:

1. CQ SV1BTR ...
2. SV1BTR K1JT ...
3. K1JT SV1BTR OOO ...
4. RO ...
5. RRR ...
6. 73 ...

For a scheduled QSO at prearranged time and frequency, transmission #1 is of course unnecessary. The ellipses (...) indicate repetition of messages, some form of which is nearly always used in EME contacts to help maximize chances of success. The "OOO" message component is a shorthand notation for a minimal signal report. It has an agreed-upon meaning that says, in effect, "your signals are readable at least some of the time, and I have copied both of our call signs." Similarly, "RO" is a shorthand message conveying both signal report and acknowledgment. It means "I have copied both calls and my signal report, and your report is O." When K1JT receives the acknowledgment "RRR" sent by SV1BTR, the QSO is complete; but since SV1BTR does not yet know this, it is conventional to send "73" or some other end-of-contact information to signify "we are done."

Shorthand radio messages have been widely employed since the days of spark and land-line telegraphy; the familiar Q-signals are another universally understood type. They are simple forms of what in communication theory is called the "source encoding" of messages. The choice of "OOO…" (repeated sequences of three carrier-on intervals separated by short spaces, with a longer space after every third one) as the signal representing a positive signal report was made by wise and experienced CW operators who knew that with extremely weak signals, "dahs" are easier to copy than "dits."

3. System Design

Fig 1 presents the flow diagram of a modern digital communication system. For maximum efficiency at low signal-to-noise ratio, a user message is source encoded into a compact form having minimum redundancy. It is then augmented with mathematically defined redundancies which can enable full recovery of the message even if some parts are subsequently corrupted by noise or signal dropouts. This process is known as "forward error correction," or FEC. The encoded message, including its error correcting information, is modulated onto a

Appendix A: Details of Message Encoding

As described in Sections §4-6, JT65 message encoding takes place in several stages. A user's message is first "source encoded" into a compact form requiring just 72 bits. The bits are packed into twelve 6-bit information symbols, and a Reed Solomon encoder adds 51 parity symbols. The 63 channel symbols are interleaved, Gray coded, and transmitted using 64-tone frequency shift keying. A synchronizing vector is sent at a 65th frequency, two tone intervals below the lowest data tone.

Some arbitrary choices define further details of message packing and the ordering of channel symbols. To make it easy for others to implement the JT65 protocol, these things are best described with actual source code examples. Appended below is a Fortran program that can easily be compiled under Linux. Only the main program is listed here; the full source code, including necessary subroutines and a Linux makefile, can be downloaded from **pulsar.princeton.edu/~joe/K1JT/JT65code.tgz**. The compiled program accepts a JT65 message (enclosed in quotes on the command line) and responds with the packed message and channel symbols as six-bit values. Examples of program output were presented in Fig 3 and described in Section §5.

```
      program JT65code


C  Provides examples of message packing, bit and symbol ordering,
C  Reed Solomon encoding, and other necessary details of the JT65
C  protocol.

      character*22 msg0,msg,decoded,cok*3
      integer dgen(12),sent(63)

      nargs=iargc()
      if(nargs.ne.1) then
         print*,'Usage: JT65code "message"'
         go to 999
      endif

      call getarg(1,msg0)                  !Get message from command line
      msg=msg0

      call chkmsg(msg,cok,nspecial,flip)   !See if it includes "OOO" report
      if(nspecial.gt.0) then               !or is a shorthand message
         write(*,1010)
 1010    format('Shorthand message.')
         go to 999
      endif

      call packmsg(msg,dgen)               !Pack message into 72 bits
      write(*,1020) msg0
 1020 format('Message:   ',a22)            !Echo input message
      if(and(dgen(10),8).ne.0) write(*,1030) !Is the plain text bit set?
 1030 format('Plain text.')
      write(*,1040) dgen
 1040 format('Packed message, 6-bit symbols: ',12i3) !Print packed symbols

      call packmsg(msg,dgen)               !Pack user message
      call rs_init                         !Initialize RS encoder
      call rs_encode(dgen,sent)            !RS encode
      call interleave63(sent,1)            !Interleave channel symbols
      call graycode(sent,63,1)             !Apply Gray code

      write(*,1050) sent
 1050 format('Channel symbols, including FEC:'/(i5,20i3))
      call unpackmsg(dgen,decoded)         !Unpack the user message
      write(*,1060) decoded,cok
 1060 format('Decoded message: ',a22,2x,a3)

 999  end
```

carrier. The resulting radio signal propagates over a channel that attenuates it, perhaps by 250 dB or more for an EME path, and adds noise as well as amplitude, frequency, and phase-changing "path modulation." Upon reception, the signal is demodulated and decoded, and the results presented to the user.

Except for the error-correcting enhancements, the flow diagram of Fig 1 describes traditional amateur CW communications just as well as modern digital techniques. In terms of the CW EME QSO outlined on the previous page, source encoding compresses the implied message "SV1BTR, this is K1JT, I have copied both of our calls" into the compact form "SV1BTR K1JT OOO." To provide some error-recovery capability and increase chances that the message will be copied, a CW operator repeats the compressed message many times during a timed transmission. To enhance the chances of copy even further, he may format the repetitions so as to transmit only calls for the first 75% of a transmission, followed by sending "OOO" repeatedly for the last 25%. He expects the receiving operator to know about these conventions, and to listen accordingly. All of these forms of source encoding help: the more that's known about the characteristics of a weak signal, the easier it is to copy. Under extremely marginal conditions, skilled operators listen for matches between what they hear and the types of message components they might reasonably expect. If a good match is found, message copy can be considered secure.

4. JT65 Source Encoding

JT65 uses exactly analogous techniques, starting out by making its transmitted messages compact and efficient. As described in the *WSJT 4.7 User's Guide*,[6] the standard "Type 1" messages of JT65 consist of two call signs, a grid locator, and an optional signal report—an enhanced form of messages 2 and 3 in the model QSO between SV1BTR and K1JT. The source encoder knows the rules by which standard Amateur Radio call signs are constructed, and uses this information to minimize the required number of information bits. An amateur call sign consists of a one- or two-character prefix, at least one of which must be a letter, followed by a digit and a suffix of one to three letters. Within these rules, the number of possible call signs is equal to 37×36×10×27×27×27, or somewhat over 262 million. (The numbers 27 and 37 arise because in the first and last three positions a character may be absent, or a letter, or perhaps a digit.) Since 2^{28} is more than 268 million, 28 bits are enough to encode any standard call sign uniquely. Similarly, the number of 4-digit Maidenhead grid locators on earth is 180×180 = 32,400, which is less than 2^{15} = 32,768; so a grid locator requires 15 bits in a message. These important ideas for the efficient source encoding of EME messages were first suggested by Clark and Karn[7] in 1996.

Any Type 1 message can be source-encoded into 28+28+15= 71 bits, plus one more for the signal report. In comparison, sending the message "SV1BTR K1JT OOO" in Morse code requires 170 bits (where a bit is defined as the key-down dot interval), even without the grid locator. The JT65 message is much more compact than the CW message, while conveying significantly more information. In practice, the JT65 protocol encodes signal reports in another way and instead uses the 72nd bit to indicate that the message contains arbitrary text instead of call signs and a grid locator. With a 43-character alphabet, the maximum plain-text message length is 13 (the largest integer less than 71 log 2/log 43). Subject to this limiting size, JT65 can transmit and receive *anything* in a message.

As indicated above, some 6 million of the possible 28-bit values are not needed for call signs. A few of these slots have been assigned to special message components such as "CQ" and "QRZ." CQ may be followed by three digits to indicate a desired call-back frequency. (If K1JT transmits on a standard calling frequency, say 144.120, and sends "CQ 113 K1JT FN20," it means that he will listen on 144.113 and respond there to any replies.) A numerical signal report of the form "-NN" or "R-NN" can be sent in place of a grid locator. The number NN must lie between 01 and 30. If required by licensing authorities, a country prefix or portable suffix may be attached[8] to one of the call signs, as in ZA/PA2CHR or G4ABC/P. If this feature is used, the additional information is sent in place of the grid locator. Some remaining details of message encoding can be found in Appendix A, and a list of supported "add-on" prefixes and suffixes is presented in Appendix B.

5. Forward Error Correction

After being compressed into 72 bits, a JT65 message is augmented with 306 uniquely defined error-correcting bits. The FEC coding rate is thus r = 72/378 = 0.19; equivalently one might say that each message is transmitted with a "redundancy ratio" of 378/72 = 5.25. With a good error-correcting code, however, the resulting performance and sensitivity are far superior to those obtainable with simple five-times message repetition. The high level of redundancy means that JT65 copes extremely well with QSB. Signals that are discernible to the software for as little as 10 to 15 s in a transmission can still yield perfect copy.

The source of this seemingly mysterious "coding gain" is not difficult to understand. With 72 bits the total number of possible user messages is 2^{72}, slightly more than 4.7×10^{21}. The number of possible patterns of 378 bits is a vastly larger number, 2^{378}, in excess of 6×10^{113}. With a one-to-one correspondence between 72-bit user messages and 378-bit "codewords," or unique sequences of 378 bits, it is clear that only a tiny fraction of the available sequences need to be used in the code. The sequences chosen are those that are "as different from one

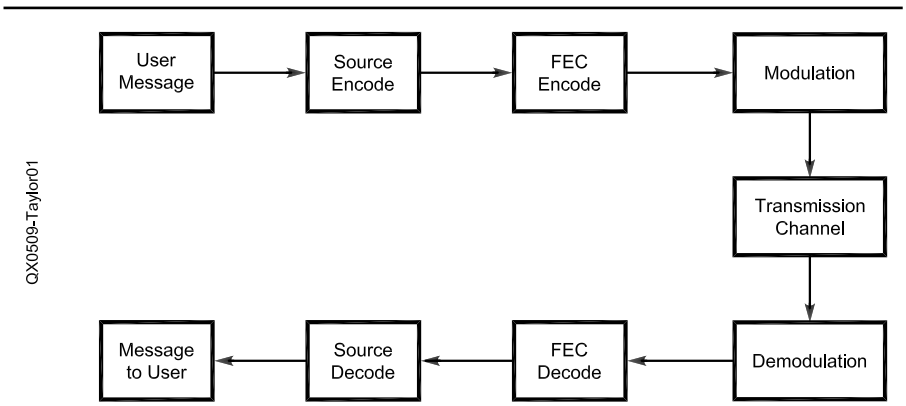


Fig 1—Schematic diagram of information flow in a digital communication system.

another as possible," in a mathematically rigorous sense.

A huge variety of efficient error correcting codes are known and understood mathematically. Among the best known are the Reed Solomon codes, used to produce the extremely low error rates characteristic of modern CD-ROMs and hard disk drives. For JT65 I chose the Reed Solomon code RS(63,12), which encodes each 72-bit user message into 63 six-bit "channel symbols" for transmission. Every codeword in this code differs from every other one in at least 52 places—which, in a nutshell, is why the code is so powerful. Even at very low SNR, distinct sequences are very unlikely to be confused with one another.

As an example, the encoded sequences for three nearly identical messages are illustrated in Fig 2. Lines labeled "packed message" show each source-encoded, 72-bit user message as a sequence of twelve 6-bit symbols. Reading from left to right, one can see that the fifth numerical symbol changes from 9 to 5 when the last letter in the first call sign changes from F to E. The final packed symbol changes from 16 to 17 when the grid locator changes from JO40 to JO41. Otherwise, the three packed messages are identical. On the other hand, the three fully encoded sequences of channel symbols appear to be almost entirely different from one another—so different that *there is virtually no chance whatsoever that, if it is decodable at all, a noise-corrupted version of one of these messages would ever be misconstrued as one of the others*. The full and exact user message has a high probability of being received, even if the key-down SNR is as low as 2 to 6 dB in 2.7 Hz bandwidth (or –28 to –24 dB in 2500 Hz, the conventional reference bandwidth used in *WSJT*. This statement can be quantified by explicit measurements of transmission error rates as a function of SNR, and such measurements are summarized for JT65 in Appendix C.

6. Interleaving and Gray Coding

After encoding, the order of JT65 symbols is permuted by writing them row-by-row into a 7×9 matrix, and reading them out column-by-column. I was studying FEC for the first time when JT65 was being designed, and I mistakenly believed that scrambling the symbol order would give the system greater immunity to signal dropouts. In fact, it does not; but since its effect is quite harmless, the procedure has been left intact to preserve the integrity of JT65 signals over subsequent program versions. The reordered symbols are converted from binary to Gray-code representation, which makes JT65 somewhat more tolerant of frequency instabilities.

7. Shorthand Messages

Like the CW methods described earlier, JT65 uses special signal formats to convey frequently used messages in a robust and efficient way. Three such messages are presently defined. They correspond exactly to the transmissions numbered 4, 5, and 6 in the model CW QSO between SV1BTR and K1JT, conveying the messages "RO," "RRR," and "73." Instead of keying a single-frequency carrier on and off according to a pattern like di-dah-dit, dah-dah-dah, …, JT65 sends "RO" by transmitting two alternating tones with specified frequencies and a specified keying rate. Such waveforms are easy to recognize and to distinguish from one another, as well as from "normal" JT65 messages. Indeed, as many users have discovered, the shorthand messages of JT65 are readily decodable by human operators using sight or sound, as well as by computer.

8. Synchronization and Modulation

JT65 uses one-minute T/R sequences and requires tight synchronization of time and frequency between transmitter and receiver. Typical amateur equipment cannot accomplish this task with sufficient accuracy in open-loop fashion, so a JT65 signal must carry its own synchronizing information. A pseudo-ran-

QX0509-Taylor02

```
Message #1:  G3LTF DL9KR JO40
Packed message, 6-bit symbols:  61 37 30 28  9 27 61 58 26  3 49 16
Channel symbols, including FEC:
   14 16  9 18  4 60 41 18 22 63 43  5 30 13 15  9 25 35 50 21  0
   36 17 42 33 35 39 22 25 39 46  3 47 39 55 23 61 25 58 47 16 38
   39 17  2 36  4 56  5 16 15 55 18 41  7 26 51 17 18 49 10 13 24

Message #2:  G3LTE DL9KR JO40
Packed message, 6-bit symbols:  61 37 30 28  5 27 61 58 26  3 49 16
Channel symbols, including FEC:
   20 34 19  5 36  6 30 15 22 20  3 62 57 59 19 56 17 35  2  9 41
   10 23 24 41 35 39 60 48 33 34 49 54 53 55 23 24 59  7  9 39 51
   23 17  2 12 49  6 46  7 61 49 18 41 50 16 40  8 45 55 45  7 24

Message #3:  G3LTF DL9KR JO41
Packed message, 6-bit symbols:  61 37 30 28  9 27 61 58 26  3 49 17
Channel symbols, including FEC:
   47 27 46 50 58 26 38 24 22  3 14 54 10 58 36 23 63 35 41 56 53
   62 11 49 14 35 39 60 40 44 15 45  7 44 55 23 12 49 39 11 18 36
   26 17  2  8 60 44 37  5 48 44 18 41 32 63  4 49 55 57 37 13 25
```

Fig 2—Three JT65 messages shown as they appear to the user; in 72-bit packed form, displayed as 12 × 6-bit symbol values; and as FEC-enhanced sequences of 63 × 6-bit channel symbols. The channel symbols are ready to be transmitted by means of 64-tone FSK, with each symbol value corresponding to a distinct tone.

dom "sync vector" is therefore interspersed with the encoded information bits. It allows accurate calibration of relative time and frequency errors, thereby establishing a rigorous framework within which the decoders can work. In addition, it enables the averaging of successive transmissions so that decoding is possible even when signals are too weak to accomplish it in a single transmission. The synchronizing signal is so important that (except in shorthand messages) half of every transmission is devoted to it.

A JT65 transmission is divided into 126 contiguous time intervals, each of length 0.372 s (4096 samples at 11025 samples per second). Within each interval the waveform is a constant-amplitude sinusoid at one of 65 predefined frequencies, and frequency changes between intervals are accomplished in a phase-continuous manner. A transmission nominally begins at t = 1 s after the start of a UTC minute and finishes at t = 47.8 s. The synchronizing tone is at frequency 1270.5 Hz and is normally sent in each interval having a "1" in the pseudo-random sequence reproduced at the top of Fig 3. The sequence has the desirable mathematical property that its normalized autocorrelation function falls from 1 to nearly 0 for all non-zero lags. As a consequence, it makes an excellent synchronizing vector.

Encoded user information is transmitted during the 63 intervals not used for the sync tone. Each channel symbol generates a tone at frequency 1270.5 + 2.6917 (N+2) m Hz, where N is the integral symbol value, $0 \le N \le 63$, and m assumes the values 1, 2, and 4 for JT65 sub-modes A, B, and C. The signal report "OOO" is conveyed by reversing sync and data positions in the pseudo-random sequence. Because normal messages depend on tight synchronization, they can be initiated only at the beginning of a UTC minute.

Shorthand messages dispense with the sync vector and use intervals of 1.486 s (16,384 samples) for the alternating tones. The lower frequency is always 1270.5 Hz, the same as that of the sync tone. The frequency separation is 26.917 nm Hz with n = 2, 3, 4 for the messages RO, RRR, and 73. By the time shorthand messages become relevant in a QSO, the frequency offset between transmitter and receiver has already been measured with high accuracy. As a consequence, these messages can be securely identified by the operator as coming from the station whose call sign was recently decoded. Accurate time synchronization is not required for shorthand messages, so

Appendix B: Supported Call Sign Prefixes and Suffixes

Call sign prefixes and suffixes supported by JT65 are listed in the file *pfx.f* included in the source code archive at **pulsar.princeton.edu/~joe/K1JT/JT65code.tgz**, as described in Appendix A. Supported suffixes include /P and /0 through /9, while the full prefix list is appended below. Additional prefixes and suffixes could be added to the list in the future. Space for 450 prefixes has been reserved by not supporting any grid locators within 5° of the North Pole.

1A	1S	3A	3B6	3B8	3B9	3C	3C0	3D2	3D2C	3D2R	3DA	3V	3W	3X
3Y	3YB	3YP	4J	4L	4S	4U1I	4U1U	4W	4X	5A	5B	5H	5N	5R
5T	5U	5V	5W	5X	5Z	6W	6Y	7O	7P	7Q	7X	8P	8Q	8R
9A	9G	9H	9J	9K	9L	9M2	9M6	9N	9Q	9U	9V	9X	9Y	A2
A3	A4	A5	A6	A7	A9	AP	BS7	BV	BV9	BY	C2	C3	C5	C6
C9	CE	CE0X	CE0Y	CE0Z	CE9	CM	CN	CP	CT	CT3	CU	CX	CY0	CY9
D2	D4	D6	DL	DU	E3	E4	EA	EA6	EA8	EA9	EI	EK	EL	EP
ER	ES	ET	EU	EX	EY	EZ	F	FG	FH	FJ	FK	FKC	FM	FO
FOA	FOC	FOM	FP	FR	FRG	FRJ	FRT	FT5W	FT5X	FT5Z	FW	FY	M	MD
MI	MJ	MM	MU	MW	H4	H40	HA	HB	HB0	HC	HC8	HH	HI	HK
HK0A	HK0M	HL	HM	HP	HR	HS	HV	HZ	I	IS	IS0	J2	J3	J5
J6	J7	J8	JA	JDM	JDO	JT	JW	JX	JY	K	KG4	KH0	KH1	KH2
KH3	KH4	KH5	KH5K	KH6	KH7	KH8	KH9	KL	KP1	KP2	KP4	KP5	LA	LU
LX	LY	LZ	OA	OD	OE	OH	OH0	OJ0	OK	OM	ON	OX	OY	OZ
P2	P4	PA	PJ2	PJ7	PY	PY0F	PT0S	PY0T	PZ	R1F	R1M	S0	S2	S5
S7	S9	SM	SP	ST	SU	SV	SVA	SV5	SV9	T2	T30	T31	T32	T33
T5	T7	T8	T9	TA	TF	TG	TI	TI9	TJ	TK	TL	TN	TR	TT
TU	TY	TZ	UA	UA2	UA9	UK	UN	UR	V2	V3	V4	V5	V6	V7
V8	VE	VK	VK0H	VK0M	VK9C	VK9L	VK9M	VK9N	VK9W	VK9X	VP2E	VP2M	VP2V	VP5
VP6	VP6D	VP8	VP8G	VP8H	VP8O	VP8S	VP9	VQ9	VR	VU	VU4	VU7	XE	XF4
XT	XU	XW	XX9	XZ	YA	YB	YI	YJ	YK	YL	YN	YO	YS	YU
YV	YV0	Z2	Z3	ZA	ZB	ZC4	ZD7	ZD8	ZD9	ZF	ZK1N	ZK1S	ZK2	ZK3
ZL	ZL7	ZL8	ZL9	ZP	ZS	ZS8								

they may be started at any time during a transmission.

By now it should be clear that JT65 does not transmit messages character by character, as done in Morse code. Instead, whole messages are translated into unique strings of 72 bits, and from those into sequences of 63 six-bit symbols. These symbols are transmitted over a radio channel; some of them may arrive intact, while others are corrupted by noise. If enough of the symbols are correct (in a probabilistically defined sense), the full 72-bit compressed message can be recovered *exactly*. The decoded bits are then translated back into the human-readable message that was sent. The coding scheme and robust FEC assure that messages are never received in fragments. Message components cannot be mistaken for one another, and call signs are never displayed with a few characters missing or incorrect. There is no chance for the letter O or R in a call sign to be confused with a signal report or an acknowledgment, or for a fragment of a call sign like N8CQ or a grid locator like EM73 to be misinterpreted. If your sked partner does not show and another station calls in his place, you will never conclude mistakenly that the schedule was kept as intended.

9. Reception and Demodulation

Within *WSJT*, a received JT65 signal is converted to baseband and analyzed using a sequence of well known DSP techniques. The process begins with an audio signal in the approximate frequency range 0-3 kHz, digitized at the nominal rate 11025 samples per second. The digital signal is low-pass filtered and downsampled by a factor of two. Power spectra are computed from discrete Fourier transforms of sliding 2048-sample blocks and examined for presence of the pseudo-random sync pattern. Detection and "peaking up" on the sync pattern establishes the required frequency and time offsets,

1,0,0,1,1,0,0,0,1,1,1,1,1,1,0,1,0,1,0,0,0,1,0,1,1,0,0,1,0,0,
0,1,1,1,0,0,1,1,1,1,0,1,1,0,1,1,1,1,1,0,0,0,1,1,0,1,0,1,0,1,1,
0,0,1,1,0,1,0,1,0,1,0,0,1,0,0,0,0,0,0,1,1,0,0,0,0,0,0,0,1,1,
0,1,0,0,1,0,1,1,0,1,0,1,0,1,0,0,1,1,0,0,1,0,0,1,0,0,0,0,1,1,
1,1,1,1,1,1

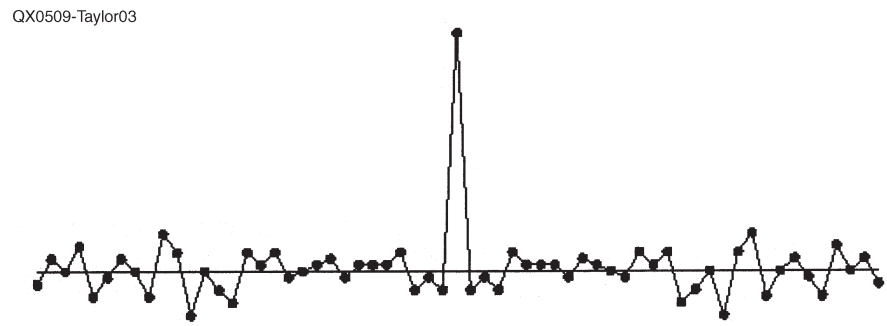


Fig 3—The pseudo-random sequence used in JT65 as a "synchronizing vector," and a graphical representation of its autocorrelation function. The isolated central correlation spike serves to synchronize time and frequency between transmitting and receiving stations.

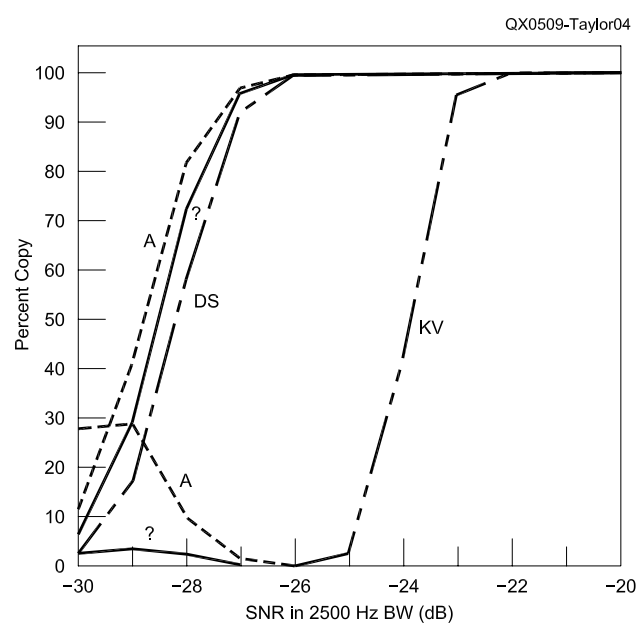


Fig 4—Measured rates of copy as a function of SNR for JT65B. The curve labeled KV refers to the Koetter-Vardy algorithm; DS refers to the deep search algorithm. The rate of false decodes for the KV algorithm is too small to measure; for the DS algorithm the rate of "hard errors" was about 0.03%, too small to show on this graph. Curves labeled "?" and "A" at the lower left give the deep-search soft-error rates for decoded messages marked "?" and when "Aggressive decoding" has been requested.

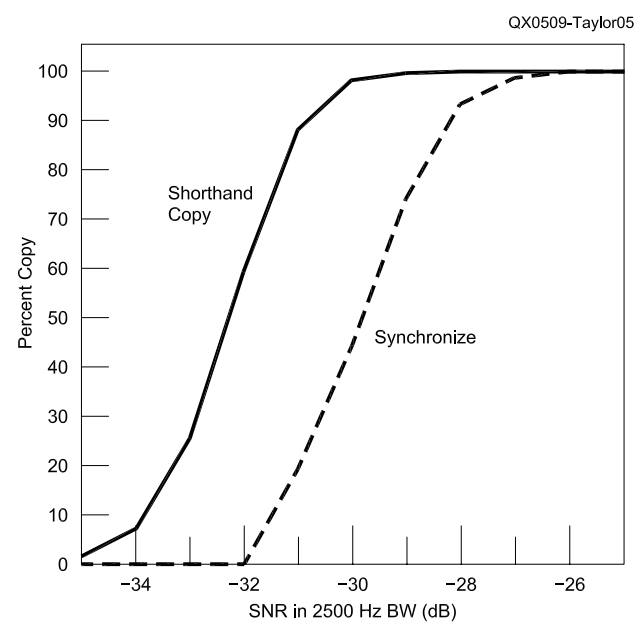


Fig 5—Rates of message synchronization and copy of shorthand messages as a function of SNR.

Appendix C: Measured Sensitivity and Error Rates

The JT65 protocol can be defined once and for all, but on-the-air performance depends on a particular software implementation of the decoder. As outlined in §9-12, version 4.9 of *WSJT* does its JT65 decoding in three phases: a soft-decision Reed Solomon decoder, the deep search decoder, and the decoder for shorthand messages. Section §13 emphasizes that in circumstances involving birdies, atmospherics, or other interference, operator interaction is an essential part of the decoding process. The operator can enable a "Zap" function to excise birdies, a "Clip" function to suppress broadband noise spikes, and a "Freeze" feature to limit the frequency range searched for a sync tone. Having used these aids and the program's graphical and numerical displays appropriately, the operator is well equipped to recognize and discard any spurious output from the decoder.

Under normal conditions in which the transmission channel can be characterized by simple attenuation, the addition of white Gaussian noise, and perhaps multiplication by a "Rayleigh fading" coefficient, the sensitivities and error rates of the decoders can be accurately measured. A software simulator for doing this was written for the *Linux* platform as the first (and very essential) part of *WSJT* program development. The simulator can generate digitized waveforms for any *WSJT* mode and inject them into band-limited Gaussian noise with a specified signal-to-noise ratio and optional fading characteristics. The resulting audio files can be saved in WAV format, then opened and decoded in *WSJT*. They can also be decoded directly within the simulator, using code identical to the *WSJT* decoder but compiled for *Linux*.

Several hundred thousand simulated JT65 transmissions have been tested in this way—first as a means of debugging and fine-tuning the decoders, and later as a way to measure the sensitivity and error rates of the finished program. Results of the simulations are summarized in Figures 4 and 5. To create Fig 4, 1000 simulated transmissions were generated and tested for each of the levels SNR = –30, –29, … –20 dB, using standard JT65 messages consisting of two call signs and a grid locator. The full *WSJT* decoder (version 4.9.5) was run on each of the 11,000 simulated transmissions. The filled circles and solid curve in Fig 4 illustrate results from the Koetter-Vardy decoder. The essential conclusion is that 96% of the transmissions were decoded correctly at –23 dB, 41% at –24 dB, and 3% at –25 dB. No false decodes were produced by the KV decoder in any of the tests.

For the deep-search algorithm, the filled squares and long-dashed curve show that 92% of the transmissions were decoded correctly at –27 dB, 58% at –28 dB, and 17% at –29 dB. Three "hard errors" (false decodes not flagged with a question mark) were recorded in the 11,000 simulated transmissions, for an overall error rate of 2.7×10^{-4} (too small to be seen in Fig 4). If one includes decoded messages flagged with a question mark, the numbers for correct copy increase to 96%, 73%, and 29% at signal levels –27, –28, and –29 dB (short-dash curve and filled triangles). The error rate, illustrated by the short-dash curve at the lower left of Fig 4, reaches a maximum of 3.6% at –29 dB. With *WSJT*'s "Aggressive decoding" option selected, the percentages of correct copy increase to 97%, 82% and 41% at –27, –28, and –29 dB (dotted curve and open triangles). However, the rate of false decodes also increases substantially, especially at –28 dB and below, reaching a maximum of 29% at –29 dB.

Similar measurements have been made for sub-modes JT65A and JT65C. The results are qualitatively similar to those shown for JT65B in Fig 4; the curves for JT65A are shifted about 1 dB to the left (more sensitive than JT65B), while those for JT65C are shifted about 1 dB to the right.

Normal JT65 messages cannot be decoded unless the sync vector is reliably detected. In *WSJT* the synchronizing procedure is exactly the same for sub-modes JT65A, B and C. For the tests illustrated in Fig 4 with SNR less than about –29 dB, failure to synchronize is the cause of many failures to decode. Synchronization is very important for another reason, as well: correct synchronization may allow the decoding of an accumulated average message, independent of whether the transmitted message is decodable with the deep search algorithm. Measured rates of synchronization are illustrated in Fig 5, again using 1000 simulated transmissions at each value of SNR over a 10 dB range. Synchronization was achieved for 93% of the test transmissions at –28 dB, 74% at –29 dB, 44% at –30%, and 19% at –31 dB. These measurements imply that message averaging should typically succeed after about 3 transmissions at –26 dB and 8 transmissions at –28 dB, but will require as many as 20 transmissions at –29 dB. These conclusions are consistent with on-the-air experience with *WSJT*.

The simulator was also used to measure the detection rates for JT65 shorthand messages, as illustrated in Fig 5. With 1000 trials at each SNR, shorthand messages were correctly decoded in 88% of the trials at –31 dB, 60% at –32 dB, and 26% at –33 dB. The total number of incorrectly decoded shorthand messages was five, in 11,000 trials. All five would have been recognized as spurious by an attentive operator, because the measured frequency offset was much larger than the normal tolerances used.

For any of a large number of reasons, on-the-air performance of JT65 may differ somewhat from the simulated results shown here. The measurements summarized in Figures 4 and 5 were made under idealized conditions with additive white Gaussian noise (AWGN) and no fading. (An additional set of simulations has been made with the effects of Rayleigh fading included; the results are qualitatively similar to those shown here, with the curves shifted several dB to the right.) The effects of birdies, other interference, and non-Gaussian noise are harder to quantify. Suffice it to say that I often leave *WSJT* running in "Monitor" mode for days at a time, with my receiver tuned to an arbitrary frequency between 144.100 and 144.160. I live in a densely populated region where plenty of birdies as well as other signals come and go on the 2 meter band. The typical rate of false decodes when monitoring a quiet band averages no more than one or two per hour. Examination of the files producing the spurious decodes nearly always reveals telltale evidence that would have caused an operator to recognize and reject the invalid information. When used in the intended manner, JT65 is a highly accurate communication protocol.

which may include Doppler shift and EME path delays as well as errors in frequency calibration and clock settings. The synchronizing accuracy is typically around 1.5 Hz in frequency and 0.03 s in time. Once "sync" has been established, the program remeasures the sync-tone frequency over small groups of tone intervals and fits a smooth curve to the results, thereby enabling the tracking and compensation of small frequency drifts. Coherent phase tracking between symbols is not required.

With accurate sync information in hand, the program computes a 64-bin spectrum for each of the 63 channel symbols. These spectra have resolution 2.7m Hz (e.g., 5.4 Hz for sub-mode JT65B, m = 2), and with very weak signals they are essentially noise-like in form. Many of the individual data tones may not be detectable above the noise. On average, however, in each tone interval the one frequency bin containing signal will have greater amplitude than the others. Using the known statistical properties of random Gaussian noise, *WSJT* computes the probability that a symbol was transmitted with each one of the possible values. This probabilistic information, based on measured spectra of the synchronized symbols, is the basic received information. After Gray coding and symbol interleaving have been removed, the probabilities are passed on to the decoder.

10. Reed Solomon Decoder

Even a small error-correcting code like RS(63,12) can be very difficult to "invert" or decode efficiently. The basic problem is this: given the measured spectra for each of the 63 channel symbols, is there a unique 72-bit sequence that can be confidently identified as the user's message? In principle, one might encode each of the 2^{72} possible user messages and correlate the results against the received spectra, looking for a match. Such an approach is quite impractical, however: a simple estimate reveals that with today's 3 GHz computer, unlimited memory, and a very efficient program, it would take about 200 million years to decode a single received message this way.

Reed Solomon codes are economically important because well defined mathematical algorithms exist for decoding them. The algorithms vary in complexity and in how closely they approach the ideal sensitivity of the method just described. Since program version 4.5, *WSJT* has used an algorithm that represents the state of the art in Reed Solomon decoding. It is based on a research paper by Ralf Koetter and Alexander Vardy,[9] and uses computer code licensed from their company, CodeVector Technologies. Furnished with soft-decision probabilistic information on received symbol values, this decoder produces a clear result for every transmission analyzed. With very high confidence, it returns either the 72 bits of the transmitted message or else a flag indicating "no result."

Error rates for the *WSJT* decoders have been carefully measured as a function of signal level. The results are summarized in Appendix C. Briefly stated, the K-V decoder exhibits a steep transition from "nearly always decoding" to "nearly always failing" as the signal-to-noise ratio decreases from about –23 to –25 dB (for JT65B) on the *WSJT* scale. The results further show that with "clean" data (additive Gaussian noise, and perhaps fading, but no interference from other signals), false decodes from the K-V decoding algorithm on RS(63,12) are so rare that you will hardly ever see one.

11. Deep-Search Decoder

What if the K-V decoder fails to produce a result? Can anything further be done? Life is too short to consider correlating all 2^{72} possible user messages in search of a match, but the number of unique messages transmitted in real EME QSOs is actually very much smaller than 2^{72}, and the ones you are most interested in are fewer still. If the more plausible and more interesting messages are tested first—more or less in the same way that one does when copying very weak CW—and if the search algorithm is instructed to "time out" if no match is found after a reasonable time, the brute-force computational approach described above can be made practical. In *WSJT*, a procedure I call the "deep search" algorithm attempts to do just this.

The deep search starts with a list of plausible call signs and grid locators. Such lists have long been maintained, both mentally and in hard copy, by most EME operators. They can be of great help when trying to determine which station might be transmitting a weak CQ, answering your own CQ, or tail-ending your last QSO. In the *WSJT* deep search decoder, each list entry is paired with "CQ" and with the home call sign of the *WSJT* user, thereby creating hypothetical test messages. If N_c calls are present in the list, approximately $2N_c$ messages will be generated, fully encoded, and the channel symbols tested for good match with the observed spectra. You can define the list of likely call signs in any way you choose. An example file is provided with *WSJT*, containing the calls of nearly 5000 worldwide stations known to have been active in weak-signal work on the VHF/UHF bands. Knowledgeable JT65 users maintain their own files, adding or deleting calls as they deem appropriate.

In effect, your call sign database defines a set of matched filters, custom designed for your station and tuned for optimum sensitivity to a subset of the messages you might reasonably expect to receive. The deep search is not sensitive to messages with call signs not in the database, or arbitrary plain text, or anything besides "CQ" or your own call in the first message field. Such messages will be decoded with the already remarkable sensitivity of the K-V algorithm. However, for any message within the defined subset, the deep search decoder provides about 4 dB more sensitivity while still maintaining a low error rate. It should be obvious that those 4 dB are essentially equivalent to the widely recognized "schedule gain" that CW operators can experience when copying familiar calls or making pre-arranged contacts.

12. Decoding Shorthand Messages

In addition to seeking a synchronizing tone modulated with the expected pseudo-random pattern, *WSJT* searches for alternating tones having the specified modulation of a JT65 shorthand message. Frequencies are measured and compared with that of the sync tone in a previous transmission, and a test is made to be sure that the modulation follows the specified square-wave cycle. If the frequencies and modulation match, and if the amplitude exceeds a preset threshold, a shorthand message detection is declared. Because of the close frequency and timing tolerances, a low detection threshold can be set while still maintaining a very low rate of false positives. Measured sensitivity curves for shorthand messages are presented in Appendix C, along with those for the K-V algorithm and the deep-search decoder.

13. Operator Responsibilities and Message Integrity

QSOs made with any of the *WSJT* modes, including JT65, require active user participation at all stages. In the presence of birdies, QRM, QRN, or other anomalies such as multipath signal distortions, operator involvement is necessary to avoid mistakes in in-

terpreting program output. Most operators find that they acquire the necessary skills easily, while making their first few JT65 contacts.

In connection with the guidelines for valid QSOs outlined in Section 2, it is worth making special mention of a particular feature of JT65. Contacts made with *WSJT* are inherently self-documenting. When a JT65 QSO is successfully completed, both operators *know* that the requisite information has been exchanged. Moreover, if desired, they have the recorded wave files to prove it. These files provide a "bit trail," an essentially incorruptible proof of copy that anyone could examine. After especially interesting or difficult QSOs, recorded waveforms and screen images are often exchanged by e-mail. I have accumulated a large library of JT65 wave files from my own QSOs, and by monitoring the bands, as well as many sent to me by others. These files have proven extremely valuable for refining *WSJT*'s algorithms for optimum sensitivity and minimum error rate, under real-world conditions. Further progress will surely be made in these areas, in years to come.

14. On-the-Air Experience

The first usable version of JT65 was finished in November 2003. Early on-the-air tests with N3FZ quickly confirmed my expectation that JT65 would become a major new weapon in the arsenal of VHF/UHF weak-signal enthusiasts. The practical advantages of error-correcting codes for weak-signal Amateur Radio communication were very plainly evident. Little wonder, I realized, that NASA always transmits its deep-space photographs back to Earth using tight source encoding and strong FEC. In deep-space communications, every decibel of improved sensitivity can save millions of dollars that would otherwise have to be spent on larger antennas or more transmitter power.

Definition of the JT65 protocol has evolved only in minor ways since the first test transmissions. Meanwhile, the decoders have been steadily improved, producing sizable advances in on-the-air performance. I have no way of knowing how many EME QSOs have been made with JT65, but the number is surely in the many thousands. Users have not hesitated to report program bugs or suggest operational improvements, and *WSJT* has greatly benefited from such feedback. A sizable new group of EME enthusiasts has sprung up, attracted by the fact that JT65 QSOs can be accomplished with much more modest setups than required for traditional methods. Hundreds of JT65 EME QSOs have been made by stations running 150 W to a single Yagi on the 2 m band, and QSOs with "big gun" stations have been made with as little as 5 W. Even 50 MHz EME QSOs, long considered among the most difficult of feats, have become a common occurrence.

15. Looking Ahead

I do not foresee the need for major revision or expansion of the JT65 technical specification. However, I can think of many ways in which the implementation of JT65 might be improved. To start with, received audio data should be processed as it comes in, rather than in "batch mode" after the whole reception period is complete. This would permit having a native real-time spectral display, and I can imagine an option to allow "early decoding" of signals after 20 or 30 s of received data have been acquired. I have learned that some sound cards exhibit errors as large as 0.6% in their sampling rates. The JT65 decoders presently in *WSJT* do not attempt to correct for such errors, and sensitivity suffers unnecessarily. A better job of detecting and suppressing interference can certainly be done. The algorithm presently used to track frequency drifts of the desired signal can be improved. Explicit tracking of Doppler-induced frequency changes is certainly desirable, especially at 432 and 1296 MHz. More accurate control of the timing of transmit/receive sequences would help, and might be possible even under *Windows*. Execution speed of the decoding procedures can be improved…and the list goes on and on. Perhaps others will take up the challenge to undertake some of these improvements, or will think of other enhancements that will be even more significant.

Note: This article has been published previously in the 39th Central States VHF Society Conference *Proceedings* (2005) and in the *Proceedings* of the 2005 Southeastern VHF Society Conference.

Notes

[1]See the *WSJT* Home Page at **pulsar.princeton.edu/~joe/K1JT**.

[2]J. Taylor, K1JT, "*WSJT*: New Software for VHF Meteor-Scatter Communication," *QST*, Dec 2001, pp 36-41.

[3]J. Taylor, K1JT, "JT44: New Digital Mode for Weak Signals," *QST*, Jun 2002, pp 81-82.

[4]The *WSJT 4.7 User's Guide* is available at **pulsar.princeton.edu/~joe/K1JT/WSJT_User_470.pdf**.

[5]C. E.Shannon, "A Mathematical Theory of Communication," *Bell System Tech. Journal*, Vol 27, pp 379-423 and 623-656, 1948.

[6]See Note 4.

[7]T. Clark, W3IWI, and P. Karn, KA9Q, "EME 2000: Applying Modern Communications Technologies to Weak Signal Amateur Operations," *Proceedings Central States VHF Society*, 1996.

[8]Call sign prefixes and suffixes were accommodated in a somewhat different way in *WSJT* versions 4.9.2 and earlier.

[9]R. Koetter, and A. Vardy, "Soft-Decision Algebraic Decoding of Reed Solomon Codes," in *Proceedings of the IEEE International Symposium on Information Theory,* p 61, 2000.

Joe Taylor was first licensed as KN2ITP in 1954, and advanced to Amateur Extra class in 1957. He has since held the call signs K2ITP, WA1LXQ, W1HFV, VK2BJX and K1JT. He earned a PhD in Astronomy and was Professor of Astronomy at the University of Massachusetts from 1969 to 1980; since then he has been Professor of Physics at Princeton University, where he also served as Dean of the faculty from 1997 to 2003. His research specialty is radio astronomy, and he was awarded the Nobel Prize in Physics in 1993 for the discovery of an orbiting pulsar. Joe currently chases DX from 160 meters through the microwave bands. You can contact him at 272 Hartley Ave, Princeton, NJ 08540-5656; **k1jt@arrl.net**. ☐☐

The Impact of Load SWR on the Efficiency of Power Amplifiers

Another detail to keep in mind while designing power amplifiers.

By Carl Luetzelschwab, K9LA

The efficiency of a power amplifier impacts the high-level system design in three ways—the power supply design, the thermal design and the mechanical design of the overall package. As the efficiency of a power amplifier increases, the power supply can be smaller, the heat sink can be smaller, and thus the overall power amplifier package can be smaller. These three issues directly impact the cost of a power amplifier.

For many years significant effort has been expended in increasing the efficiency of power amplifiers. Power amplifier designers have gone from accepting the performance of Class C, Class B, and Class AB power amplifiers in the old days to a myriad of classes of operation to improve efficiency—Class D, Class E, Class F, Class E/F, Class S, several others, and variations thereof.[1] Additionally, much work has been done in efficiency enhancement techniques: the Doherty amplifier, Chireix's outphasing amplifier, envelope elimination and restoration, and variations of these techniques have been employed.[2] All of these improvements in efficiency depend upon a carefully controlled load for the amplifier.

In a laboratory environment, achieving high efficiency in a power amplifier is relatively easy. This is so because the power amplifier is usually working over a small range of frequencies (the 40-m CW band, for example) into a resistive load ($50 + j0\ \Omega$). In the real world, though, the amplifier may have to work over a wider range of frequencies (most of the author's power amplifier experience is from 30-512 MHz) and the load is a real-world antenna that can have an SWR quite a bit greater than 1:1 (for example, the power amplifiers used in the aforementioned 30-512 MHz frequency range are typically required to drive an antenna SWR as high as 3:1). The purpose of this article is to show how the efficiency of a power amplifier varies as the load SWR phase angle varies.

The effect of variable load SWR will be examined through computer modeling. To begin, the modeled performance of a power amplifier driving a 50-Ω load will be compared to the measured data of the actual power amplifier driving 50 Ω to gain confidence in the model. Then the model will be used to assess the impact of load SWR on the drain efficiency of the power amplifier. Finally, techniques to mitigate efficiency degradation due to load SWR will be discussed.

The Power Amplifier

The power amplifier modeled was the one that appeared in the January/February 2004 issue of *QEX*.[3] It is a 200 W Class E/F high-efficiency design using a pair of International Rectifier IRFP044N power MOSFETs. It is designed for 40-m CW and a nominal 12.8 V power supply. From Fig 5 in the *QEX* article, the power amplifier delivered 200 W to a 50-Ω load at 5 W input (16 dB gain) at a drain efficiency just under 80%.

[1]Notes appear on page 17.

1227 Pion Rd
Fort Wayne, IN 46845-9510
k9la@arrl.net

The Model

The modeling was done using harmonic balance simulation in Agilent Technologies *Advanced Design System (ADS) 2003C* software. The model of the power amplifier driving a 50-Ω load is shown in Fig 1.

The basic schematic of the power amplifier model came from Fig 3 in the *QEX* article. There were some errors in the original *QEX* article, and these were corrected in subsequent issues (Mar/Apr 2004, page 61, May/June 2004, page 61-62 and Jul/Aug 2004, page 61).

The model of each IRFP044N MOSFET came from International Rectifier's Web site.[4] The IRFP044N model is a *SPICE* model, and it was imported into *ADS* without difficulty. The gate bias on the IRFP044N MOSFETs was set to 3.5 V, as given in the *QEX* article.

Parameters for the passive components mostly came from the *QEX* article (and the subsequent corrections) and from the respective component data sheets. If data was not given in the article or was not available from vendor data sheets (for example, inductor loss data), it was estimated based on the author's experience with power amplifiers.

A frequency of 7.025 MHz was chosen for the model.

ADS Data Display Screen

Fig 2 is a typical *ADS* data display screen with several pertinent parameters displayed.

The two upper panels show the drain voltage pulse at each IRFP044N. Note that the two IRFP044Ns are operating in push-pull (the time difference between the first pulse in the FET 1 Voltage curve and the first pulse in the FET 2 Voltage curve is 71 nsec, which corresponds to 180° at 7.025 MHz).

The two middle panels show the drain current pulse at each IRFP044N. Note that there is very little overlap in the drain current pulse and the drain voltage pulse (from the upper panel) for either transistor. In other words, the drain current pulse (I) and the drain voltage pulse (V) do not occur simultaneously. With a minimal I×V product across the transistor at any given time, the dissipative loss in either IRFP044N is low and the result is high efficiency.[5]

The bottom left panel is the voltage across the load (Term2 in Fig 1). It is used to calculate the power into the load. The m7 marker is a peak voltage, so it has to be multiplied by 0.707 to calculate the rms voltage. This is then used to calculate the average power delivered to the load.

The bottom right panel shows the supply current (marker m2) from the 12.8 V source (SRC1 in Fig 1). This is used to calculate the drain efficiency of the power amplifier using the equation:

Drain efficiency = (average power delivered to load) / (12.8V × supply current).

Performance— Modeled versus Measured

In the model, the input power P_{in} was varied from 1 W to 20 W, and the simulated output power P_{out} and simulated drain efficiency were recorded. This data is plotted in Figs 3 and 4,

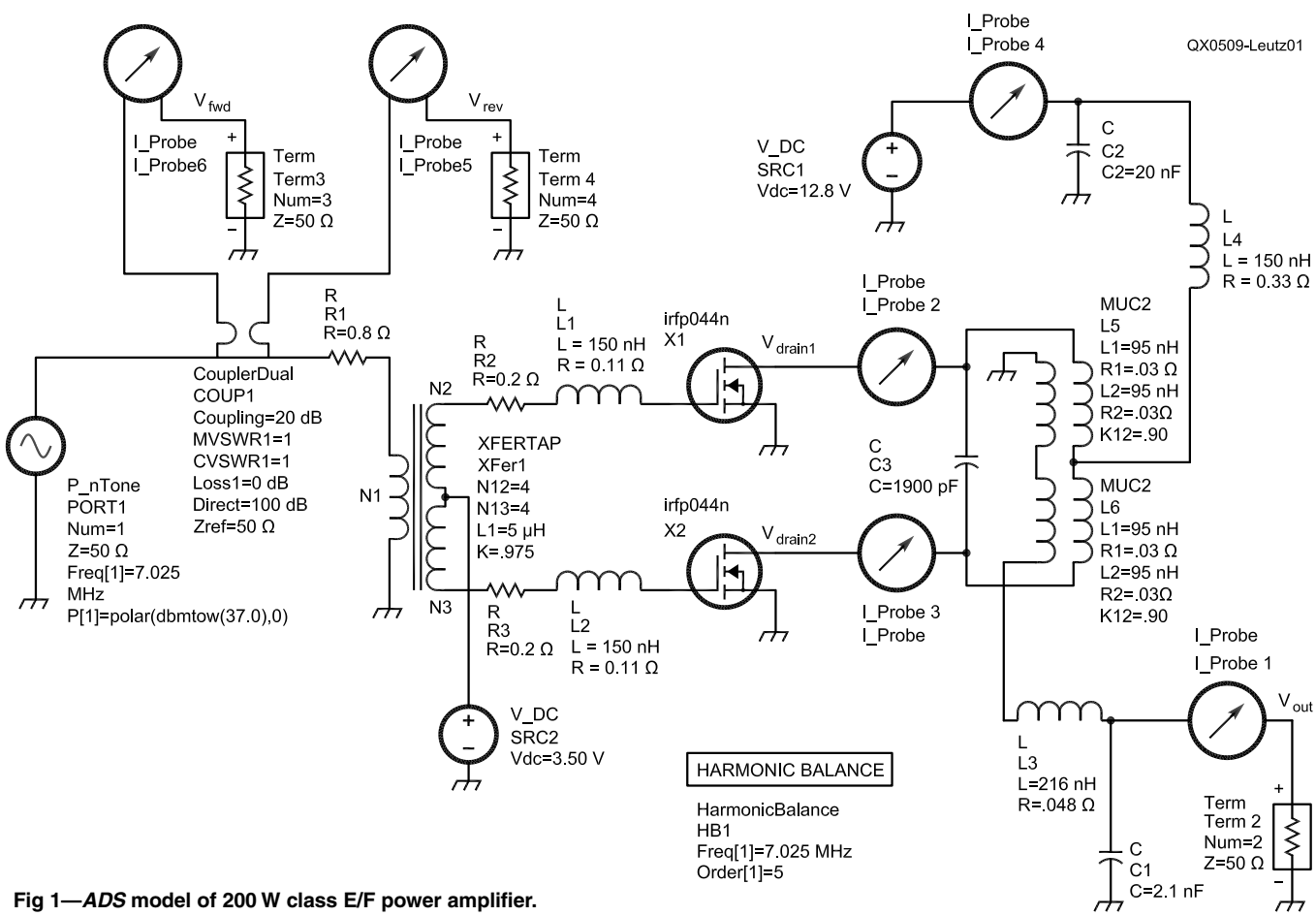


Fig 1—*ADS* model of 200 W class E/F power amplifier.

Table 1—Baseline 50 Ω Performance (Model)

Drain voltage = 12.8 V

Load impedance	*Drain current*	*Drain efficiency*	*Power Delivered to Load*	*Power Dissipated (waste heat)*	*IRFP044N Flange Temperature*	*IRFP044N Junction Temperature*
50 + *j*0 Ω	20.1 A	77.7%	200 W	57 W	50° C	68° C

Push-pull operation.
Junction-to-case thermal resistances = 1.3° C/W (from IRFP044N data sheet).
Case-to-sink thermal resistance = 0.24° C/W (from IRFP044N data sheet).
Sink-to-ambient thermal resistance = 1.5° C/W (3×3 inch heat sink with 1.5 inch long fins using 250 ft/min forced air).
Ambient temperature = +25°C.
50% duty cycle (from *QEX* article).

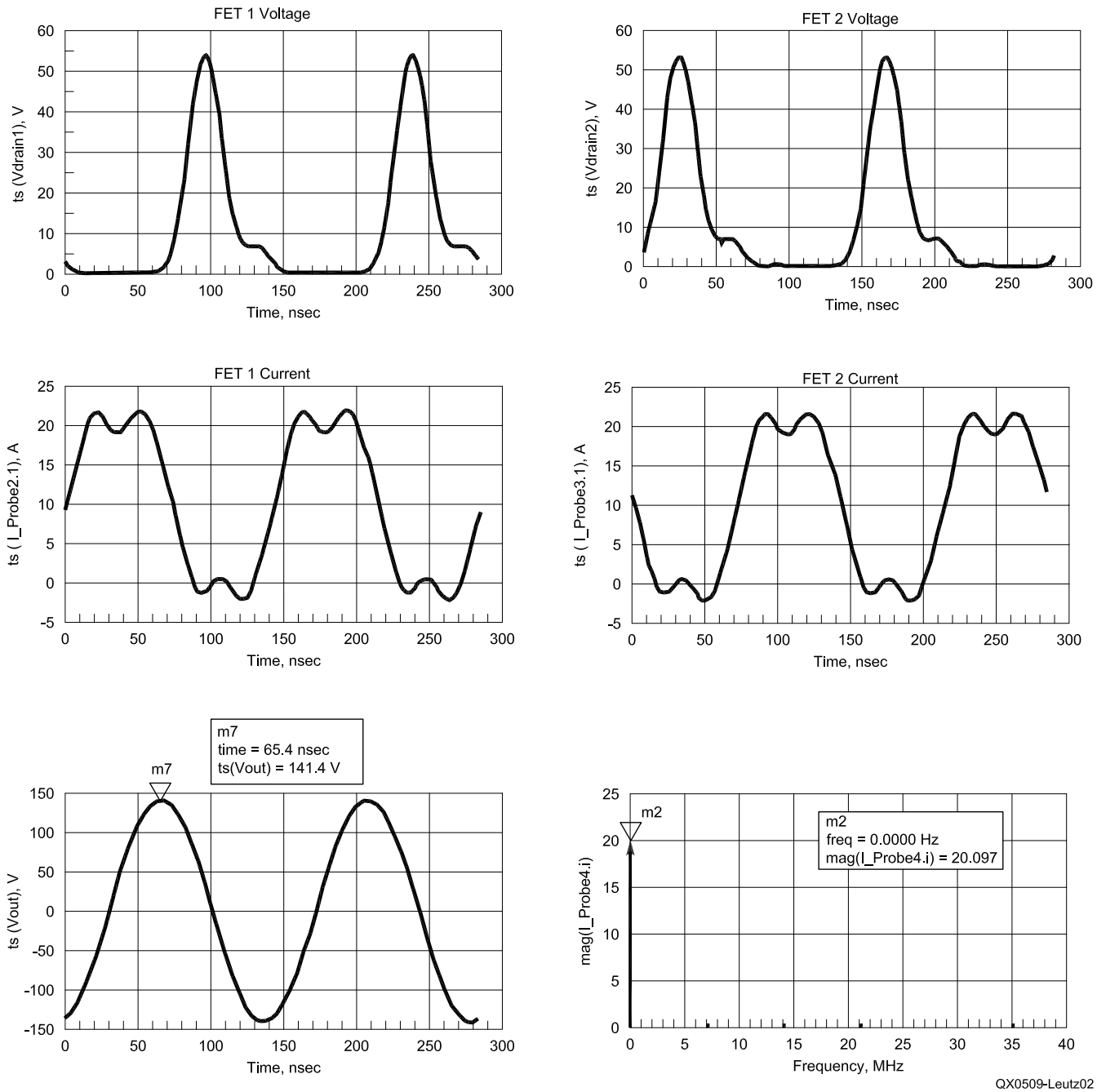


Fig 2—Typical *ADS* screen.

along with the measured data from the *QEX* article.

As can be seen in Figs 3 and 4, the modeled P_{out} versus P_{in} curve and the modeled Drain Efficiency versus P_{in} curve compare favorably to the measured data.

The 50-Ω data taken from the model is shown in Table 1, and serves as a reference for the SWR data that follows. The assumptions used to calculate the data in Table 1 follow the table.

With assurance from Figs 3 and 4 that the model satisfactorily emulates the actual power amplifier, operation into a mismatched load was simulated next.

Choosing a Practical Load SWR

Since the high efficiency *QEX* power amplifier was designed for 40-m CW, a horizontal 40-m dipole at 50 feet over average ground using #12 wire was modeled. The antenna was designed to resonate at the center of the 40-m band (7.150 MHz). Analysis using NEC 2.0[6] yields a resistance at resonance of 86.6 Ω, which is a SWR of 1.73:1. The SWR rises to around 2:1 at each end of the band.

Since the *QEX* power amplifier was designed and modeled for 40-m CW at the low end of the band, the SWR used to evaluate the power amplifier efficiency will be 2:1 at all phase angles (to emulate any length of transmission line from the dipole to the power amplifier). Loss in the length of transmission line is ignored. In real HF applications the loss would be small and would slightly reduce the SWR seen by the power amplifier.

This analysis is similar to a "load pull" measurement often performed on real hardware to learn its response to a non-50 Ω load in which the phase angle is varied.

Effect of Load SWR

The power amplifier performance into a 2:1 SWR was simulated by using a load with a reflection coefficient magnitude of 0.333 at eight discrete phase angles: 0°, –45°, –90°, –135° –180°, +135°, +90° and +45°. The infinite number of phase angles necessary to simulate *any* length of transmission line is reduced to only eight discrete angles separated by 45° to simplify a very time consuming exercise. The results are more than sufficient to illustrate the effect of varying the SWR phase angle. The input power was held constant at 5 W. The result of this exercise is shown in Table 2. The column "Power Delivered to the Load" is the power into the resistive part of the load impedance.

Figs 5 and 6 are plots of the data in Table 2 and illustrate the sinusoidal nature of the various amplifier parameters as the 2:1 load phase angle presented to the power amplifier was varied.

This sinusoidal nature is typical of what is seen with practical power

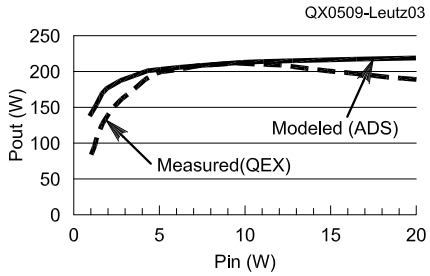


Fig 3—Comparison of P_{out} vs P_{in}.

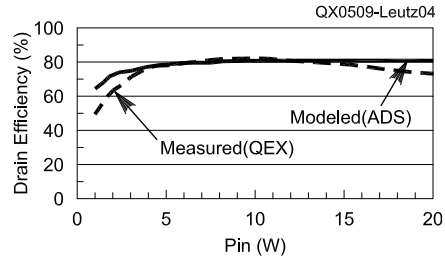


Fig 4—Comparison of efficiency vs P_{in}.

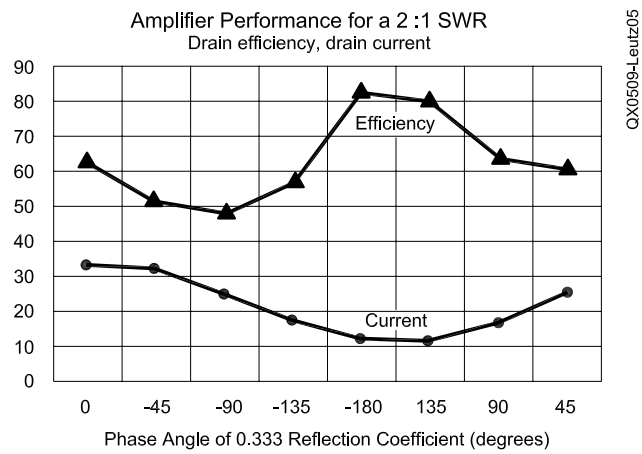


Fig 5—Drain efficiency and current.

Table 2—Modeled Performance into a 2:1 SWR

Reflection Coefficient	*Load Impedance*	*Drain Current*	*Drain Efficiency*	*Power Delivered to Load*	*Power Dissipated (waste heat)*	*IRFP044N Junction Temperature*
0.333∠0°	100 + *j*0	33.1 A	62.3%	264 W	160 W	147° C
0.333∠–45°	70 – *j*37	32.1 A	51.6%	212 W	199 W	176° C
0.333∠–90°	40 – *j*30	24.8 A	48.2%	153 W	164 W	150° C
0.333∠–135°	28 – *j*15	17.8 A	56.6%	129 W	99 W	100° C
0.333∠–180°	25 + *j*0	12.2 A	82.6%	129 W	27 W	46° C
0.333∠+135°	28 + *j*15	11.6 A	80.1%	119 W	29 W	47° C
0.333∠+90°	40 + *j*30	16.8 A	63.7%	137 W	78 W	84° C
0.333∠+45°	70 + *j*37	25.5 A	60.7%	198 W	128 W	122° C

amplifiers. The minimum-to-maximum excursion about the baseline of each parameter will depend on the specific power amplifier design.

Table 2 and Figs 5 and 6 show several interesting results. First, as expected, the drain efficiency varies significantly as the power amplifier is presented different impedances around a 2:1 SWR. Second, the power delivered to the load also varies significantly as the power amplifier is presented different impedances around a 2:1 SWR. Third, as a result of both the efficiency changing as the load goes around a 2:1 SWR circle and the amount of RF power produced as the load is varied, the IRFP044N junction temperature varies by more than 125°C.

There are two important issues revealed by the data in Table 2 and Figs 5 and 6, and both are the result of a load on the high impedance side of the 2:1 SWR (a reflection coefficient angle between roughly 0° and –45°). First, the current drain could easily exceed the rating of components in the power amplifier and of components in the power supply (the IRFP044Ns are rated at 53 A continuous, so the modeled 33 A shouldn't be a problem). Second, the calculated junction temperature of the IRFP044N is right at its maximum junction temperature rating (175°C from the IRFP044N data sheet).

To get around the high current drain issue, a bigger power supply could be designed or current limiting could be employed in the power supply design. The junction temperature issue is probably the lesser of two evils, as exceeding the 175°C rating *by a small amount* is not a *fall off the cliff* issue—it's more of a long-term reliability issue. But remember that the calculated junction temperature of 176° C is for a duty cycle of 50% (from assumption 7 after Table 1). If the power amplifier were used in a data mode (RTTY, for example) at the worse case 2:1 SWR, the junction temperature would be around 327°C. This is a serious problem, and something would need to be done.

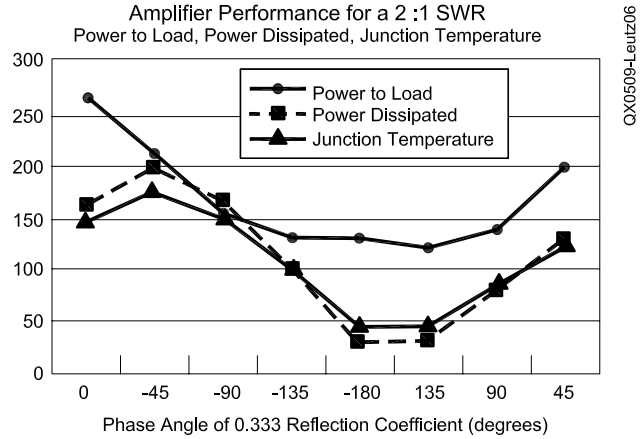


Fig 6—Power to load, power dissipated, junction temperature.

Mitigating Current Drain and Junction Temperature Issues

Although using this power amplifier at a worse case 2:1 SWR at 50% duty cycle shouldn't cause any catastrophic failures, there are two approaches to mitigating increased current drain and increased junction temperature that would keep performance more constant when operated into an SWR.

If the use of a power amplifier is confined to narrow-band operation (for example, 40-m CW), then it would be a simple matter to prune the antenna to the desired frequency. In the case of the aforementioned horizontal dipole cut for 7.150 MHz, the antenna could be physically lengthened and the wires sloped down at about a 45° angle (making it into an inverted V). This will move the resonance down to the CW portion of the 40-m band and it will also decrease the resistance at resonance to very near 50 Ω—just what the high efficiency power amplifier was designed for.

If the power amplifier is used over a wider bandwidth or on several bands where the antenna SWR is a compromise, then the mitigating approach would be to use an antenna tuner to always present 50 + *j*0 Ω to the power amplifier. This would also be the *required* solution if an ALC (automatic level control) loop is put around the power amplifier to keep its output power more constant versus antenna impedance.

If the power amplifier were for VHF and above, an isolator (or a circulator with an external 50-Ω load) would be an effective method to mitigate the effects of SWR.

Conclusion

This article demonstrates that the efficiency of a power amplifier varies significantly when it is operated into a moderate SWR having an arbitrary phase angle. This degradation could place a current drain stress on the power amplifier components and power supply components including the possibility of creating excessive junction temperatures in the power amplifier active devices.

The solution is to restrict operation to a narrow band over which the antenna has been optimized, use an antenna tuner, or use an isolator (power amplifiers for VHF and above).

Acknowledgment

I'd like to thank Ed Paragi, WB9RMA, for his review of and suggestions to this article.

Notes

[1]Peter B. Kenington, "High-Linearity RF Amplifier Design," Artech House, 2000.

[2]Steve C. Cripps, "RF Power Amplifiers for Wireless Communications," Artech House, 1999.

[3]T. Taniguchi, K. Potter, KC6OKH, and D. Rutledge, KN6EK, "A 200 W Power Amplifier," *QEX*, Jan/Feb 2004, pp 3-6.

[4]**www.irf.com/product-info/models/SPICE/irfp044n.spi**

[5]It's important to note that the high efficiency of the *QEX* power amplifier is not because it's operating in push-pull. It's due to shaping of the drain waveforms on each transistor to minimize any overlap between the drain voltage pulse and the drain current pulse on each transistor.

[6]K6STI's *NEC/WIRES 2.0* software was used to simulate the dipole. W7EL's *EZNEC* software could also be used. □□

A USB Controlled VFO

Diverse technologies meet in this DDS VFO with a 6CL6 buffer.

By Gary Geissinger, WAØSPM

After reading "Implementing a USB Equipment Interface Using the Microchip PIC16C745" by Dick Lichtel, KD4JP, in the May/June 2004 issue of *QEX*, I decided to try my hand at building a USB controlled project. Having just repaired a Heathkit DX-60B transmitter, I decided to build a USB-controlled VFO to go with it. Just for fun, I decided to make this project a truly "mixed-signal" design and use a vacuum tube along with a microcontroller and a DDS chip. The schematic is shown in Fig 1. The chips and tube are visible in Figs 2 and 3.

Digital Globe Inc.
1601 Dry Creek Dr, Ste 260
Longmont, CO 80503
ggeissinger@digitalglobe.com

Design Decisions

Several approaches were considered. The first was the user tuning interface. Incremental encoders are much more "user friendly" than other mechanisms, such as "up/down" buttons, a touch sensitive screen or a mouse. Of course, an encoder will move with vibration, so a "Tune/Lock" switch would be required. On the other hand, PCs have great display

Table 1—Periodically Repeated Actions

PC: Compute the DDS tuning words for the current transmit frequency.
PC: Write the tuning words to the VFO.
VFO: Receive the tuning words and load them into the DDS chip.
VFO: Read the encoder change register and the status bits.
VFO: Send update to the PC.
VFO: Zero the encoder change register.
PC: Read the status bits, update the frequency, keep the frequency in band.

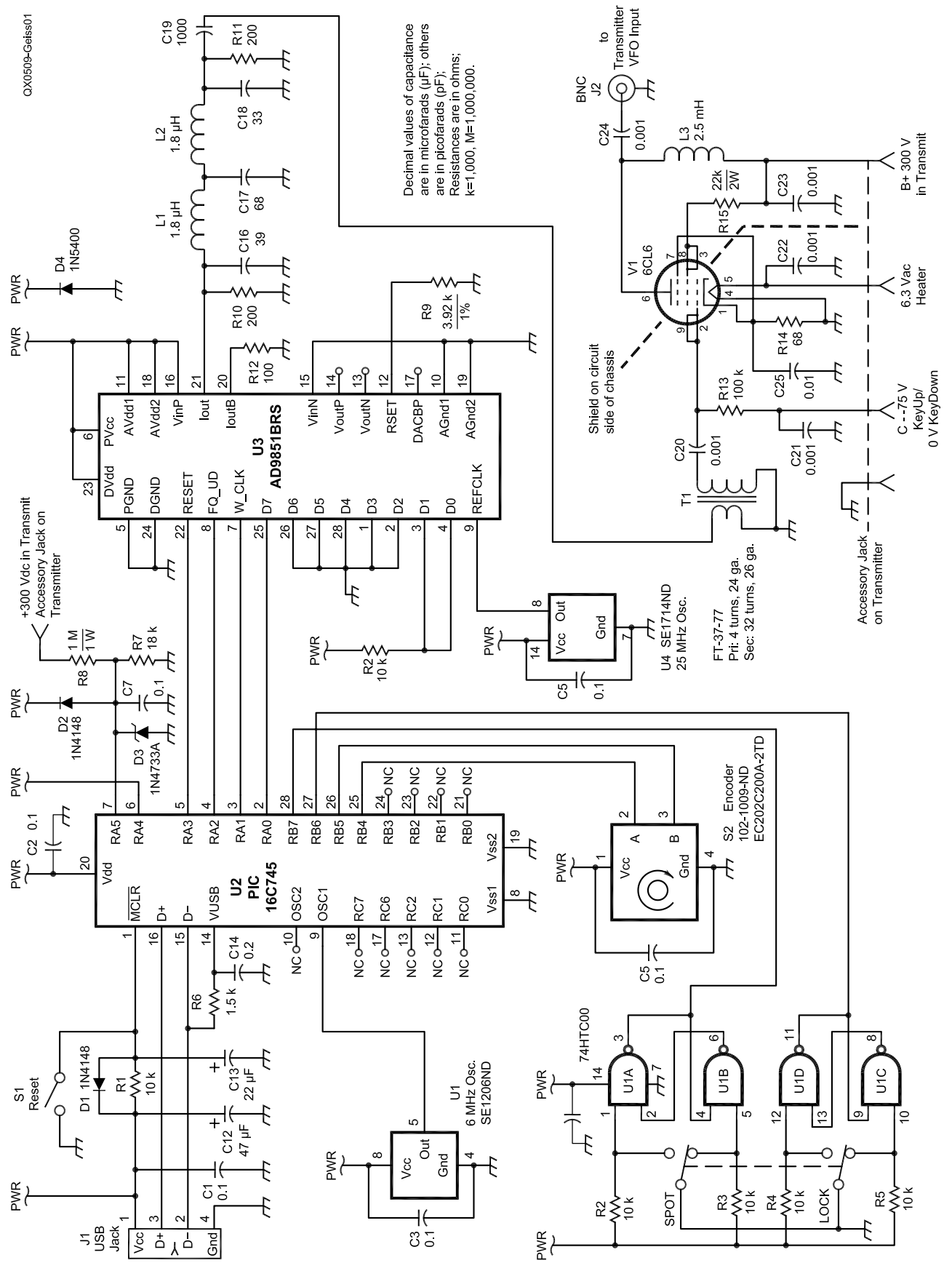


Fig 1—A schematic diagram of the USB controlled VFO.

and computational capabilities, so a PC screen was used as the display, with all computations being performed in the PC itself. By powering the digital circuitry from the USB port, we can guarantee that the VFO doesn't operate in an undefined mode—with no PC present. The PC code was designed to guarantee that all transmissions occur within the 80, 40, 20, 15 and 10 meter ham bands. Finally, a text file on the PC is used to restore the band, frequency and tuning-step settings from the last time the VFO was used.

Design Philosophy

The Borland *C++ Builder* (version 6.0) environment was used for the PC software. This software has a programmable timer function, so that actions can be periodically repeated. In this case, I chose 50 ms. This means that the code in Table 1 executes each 50 ms.

In the background, the microcontroller in the VFO counts encoder changes and stores them in a register. An assembly language programmed Microchip PIC16C745 provides local control of the VFO.

Circuit Description and Construction—VFO Digital Board

The PIC16C745/JW microcontroller interfaces the DDS chip to the USB connection. The microcontroller does a hard reset when power is applied, and whenever S1 is pressed. Except during debugging and testing, S1 is never used, so it is mounted inside the enclosure on the board.

In addition to the USB port, the microcontroller has other input and output ports. Four output bits (RA0—RA3) send tuning words to the DDS chip. RA5 samples the +300 V from the transmitter so the DDS can be inhibited when the VFO B+ voltage isn't present. This way, any signal leakage from the DDS chip won't generate a constant signal in a receiver. If you want a spot signal, use switch S2 to manually enable the DDS chip via the RB7 bit on the microcontroller. Bit RB6 allows for the tuning to be locked when S3 is actuated, so that vibration won't change the transmit frequency. Both S2 and S3 are debounced by U4. While this is not necessary in the final design, it was done in case the "interrupt on change" capability of the microcontroller was ever implemented. S2 and S3 are mounted on the front panel as shown in Fig 4. Fig 5 shows the rear panel.

Bits RB4 and RB5 accept the tuning pulses from the incremental encoder. These signals are two square-waves phased 90° apart. Frequency changes are determined by the encoder pulses (200 pulses per revolution), while the lead/lag relationship of the two pulse trains indicates the direction of rotation and of frequency change.

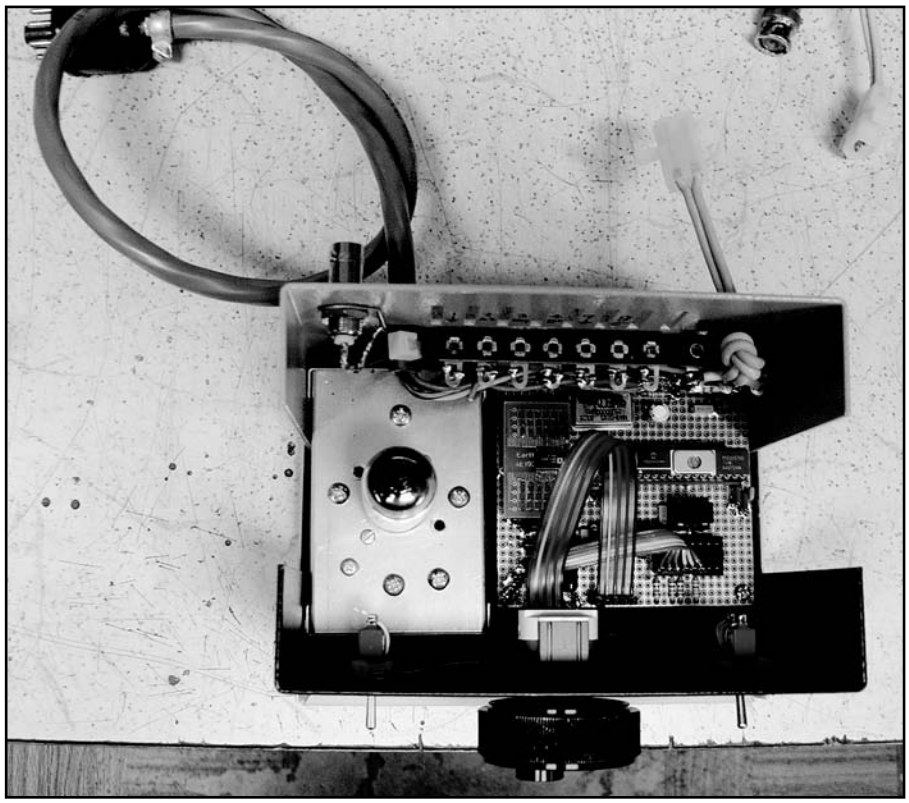


Fig 2—A top-view photo of the VFO with the vacuum tube at left-center. The knob, S2 and S3 are visible at the bottom.

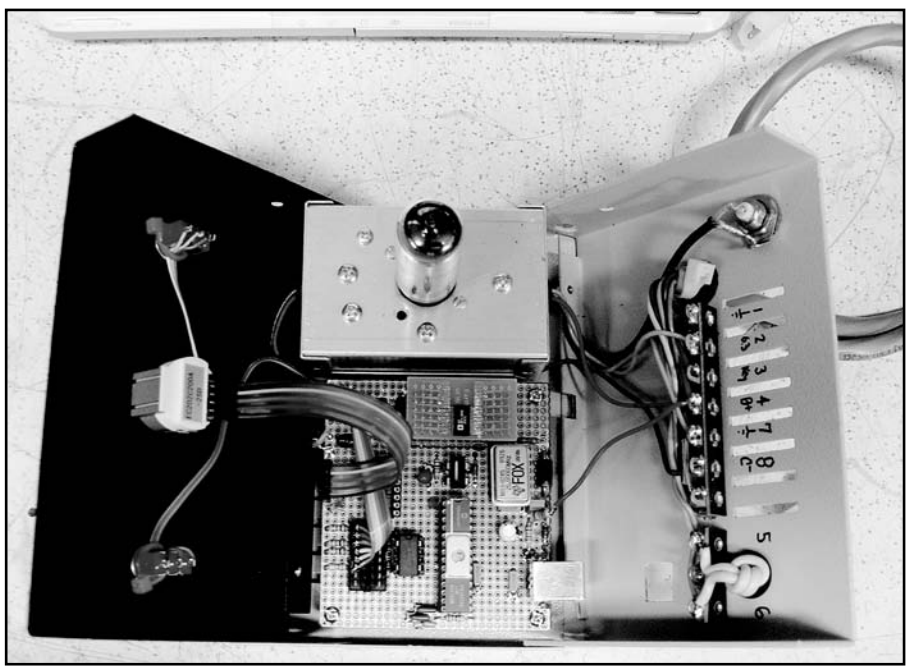


Fig 3—A photo of the "exploded" VFO. The front panel is at left and the rear panel at right.

U2, an Analog Devices AD9851BRS, is the DDS chip. It has an internal ×6 multiplier, so the 25 MHz input clock allows the DDS to update at a 150-MHz rate. The internal accumulator is a 32-bit register. The 32-bit tuning word that is updated by the microcontroller is added to this accumulator 150 million times a second. So, the output frequency of the DDS chip is determined by:

$$f_{out} = 150MHz \frac{tuning\ word}{2^{32}} \quad \text{(Eq 1)}$$

Several more design issues need to be considered. First is the design of the DX-60B transmitter. This transmitter is designed to use 80 m (3.5-4.0 MHz) and 40 m (7.0-7.3 MHz) crystals. For the 20, 15 and 10 meter bands, the transmitter multiplies the crystal (or VFO) frequency by 2, 3 or 4. As a result, the required DDS output frequency for the 20, 15 and 10 m bands is actually in the 7 MHz region. Second, it is unlikely that the 25 MHz oscillator is *exactly* on frequency. Therefore, the PC software has a 25 MHz constant that can be adjusted to calibrate the VFO. This sort of "soft" calibration is much cheaper than buying a precision 25 MHz oscillator!

The output of the DDS chip is low-pass filtered by a two-section π filter. The input and output capacitances need to be near 39 pF. The RG-174 coaxial cable from the output of the DDS board to the input of the tube buffer amplifier has about 6 pF of capacitance; hence, the output capacitor is reduced to 33 pF from 39 pF. This cable must be kept as short as practical.

The digital board is built on two-sided 0.1-inch centers "pad per hole" board from Twin Industries. Between the pads, there is a plane on each side of the board. The top plane (component side) is assigned to be ground; the bottom plane (wiring side) is assigned to be +5 V dc power. Connections to the power and ground plane are made using T124 solder washers from Vector Electronic Company. By placing the solder washer over the part or socket lead on one side or the other, a tight, low inductance power or ground connection can be made.

Two different sizes of solder washers are used. All discrete components are connected to either power or ground using 0.040 inch ID solder washers. For example, a 0.l μF bypass capacitor between power and ground is connected to the board using a solder washer on the top plane on one lead and another solder washer on the bottom plane on the other lead. Using this approach, the lead inductance can be reduced almost to zero.

By the way, there should be a 0.1 μF monolithic capacitor mounted very close to every IC power and ground lead pair. The DDS chip has four pairs; as a result, you need to use four capacitors. Don't skimp here; the capacitors are cheap.

Low-profile wire-wrap sockets and socket strip were used to hold ICs. For the low-profile sockets, connections to ground (top side) are made with 0.058 ID solder washers, while power connections (bottom side) use the 0.040 ID solder washers. A Dremel tool was used to grind away the planes when necessary; for example, the stand-offs that mount and ground the board to the metal enclosure would short power and ground together if the power plane wasn't removed around the holes for the standoffs.

Note: Solder washers are usually sold in quantities of 500 or 1,000 (pretty pricey!). If you don't use them routinely, it is possible to connect pins and component leads to planes using "solder globs." While I don't recom-

Fig 4—A front view of the VFO. The spot switch (S2) is at left and the lock switch (S3) is at right.

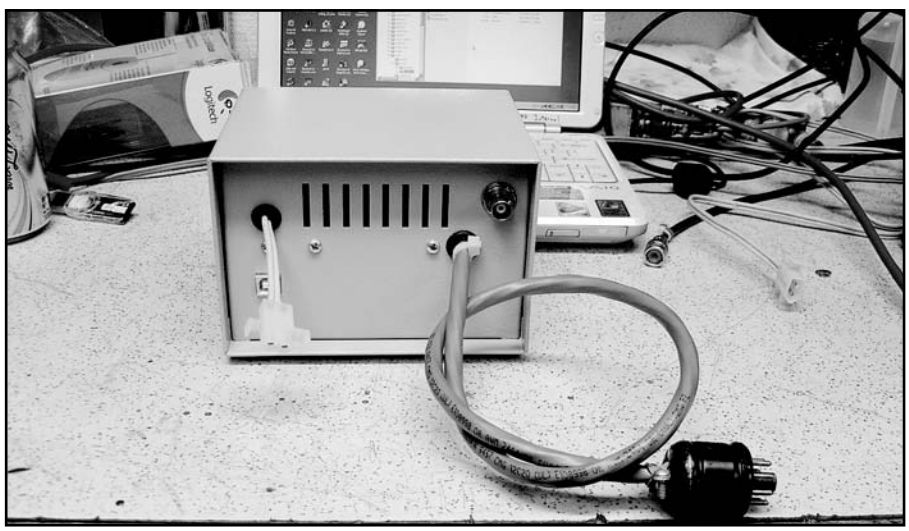


Fig 5—A rear-view photo of the VFO.

mend this, I know of at least one ham who has good luck using this connection approach.

Signal wiring was done using #30 AWG wire-wrap, although connections could be made with soldered leads if one is careful. The layout and the wiring must keep leads short and direct. This is especially true for the 25-MHz and 6-MHz oscillator signals.

The AD9851BRS chip is a 28-pin SOIC surface-mount part. A socket adapter is needed to convert the surface-mount format into one on 0.1-inch centers, to be compatible with wire-wrap socket strip. While I designed my own adapter board, the ones available from Emulation Technology (and no doubt others) will work just fine.

Circuit Description and Construction: Vacuum-Tube Amplifier

The amplifier was built using conventional point-to-point wiring in an aluminum box. A shield added in the aluminum box isolates the input and output circuits. This is necessary as the 6CL6 has considerable power sensitivity and might oscillate without the shield.

The 6CL6 was chosen because it runs well on the voltages supplied by the DX-60B, has a nice, low plate resistance and has good gain. Other tubes, such as the 5763, 6417 and 12BY7 should work just fine. The low plate resistance allows for minimal loss down the coaxial cable that sends the RF signal from the tube buffer into the VFO INPUT jack on the transmitter. Once again, use the shortest practical cable.

The input matching transformer is wound on an Amidon FT-37-77 core. Some experimentation may be required if you change the layout or choose a different tube.

Apart from the "fun factor" of using a tube, there is a legitimate technical reason for using a tube rather than a solid-state circuit. The DX-60B supplies all of the voltages needed for tube circuitry, including a keyed C– voltage. When this is fed to the control grid of the 6CL6, the amplifier is cut off during key-up conditions. In this way, signal leakage during key-up periods is further reduced.

Software

The PIC16C745 in the VFO was programmed in assembly language using *MPLAB*. This software environment is available at no charge from Microchip (**www.microchip.com**). The actual program is a modified version of the code written by KD4JP. Only the main program file was modified to perform in this application. The fuse file that I generated is available, along with the rest of the files, from the *ARRLWeb*.[1]

[1]You can download this package from the ARRL Web at **www.arrl.org/qexfiles/**. Look for 9X05Geissinger.zip.

The assembled machine code was programmed into the microcontroller using a PIC16PRO programmer board. While mine was procured from a surplus store for $25, the circuit board and software is furnished with the textbook, *Programming and Customizing PICmicro Microcontrollers* by Myke Predko. (2nd edition, McGraw-Hill, ISBN 0-07-136172-3.) Several Web vendors supply an enhanced version of the software required to program the PIC16C745. Don't forget to program the configuration register! (Hint: If you ever need to erase this part and don't have an EPROM eraser, a UV "mineral lamp" held 1 inch from the chip for 20 minutes will do the trick! Mine is made by "UV Products" and uses a number 34 0003 01 tube.) Beware: Some mineral lamps emit a different UV spectrum that won't erase the parts.

The microcontroller functions over USB as a "Human Interface Device" or HID. In this implementation, each transaction has 8 bytes of data attached. The PC sends tuning words to the microcontroller, and the microcontroller returns status bits and the encoder tuning change information.

The Borland *C++ Builder 6.0* environment on the PC doesn't allow the user to write USB control software conveniently. Two approaches were used to work around this; both work just fine.

The first approach used a pur-

Table 2—USB Controlled VFO Parts List

part—Value/part number
C1-C11—0.1 µF, 35 V monolythic ceramic capacitor
C12—47 µF, 16 V tantalum capacitor
C13—22 µF, 16 V tantalum capacitor
C14—0.2 µF, 35 V monolythic ceramic capacitor
C16—39 pF silver mica capacitor
C17—68 pF silver mica capacitor
C18—33 pF silver mica capacitor
C19—1000 pF silver mica capacitor
C20-24—0.001 µF, 1 kV disc ceramic capacitor
C25—0.01 µF, 500 V disc ceramic capacitor
D1, D2—1N4148 silicon signal diode
D3—1N4733a 5.1 V Zener diode
D4—1N5400 power Diode
J1—ae1085ND USB jack (Digikey)
J2—BNC jack, panel mount
L1, L2—1.8 µH miniature inductor
L3—2.5 mH choke (Antique Electric Supply)
R1-R5—10 k, 1/4 W resistor
R6—1.5 kΩ, 1/4 W resistor
R7—18 k Ω, 1/4 W resistor
R8—1 MΩ, 1 W resistor
R9—3.92 kΩ, 1/4 W, 1% resistor
R10, R11—200 Ω, 1/4 W resistor
R12—100 Ω, 1/4 W resistor
R13—100 kΩ , 1/4 W resistor
R14—68 Ω, 1/4 W resistor
R15—22 kΩ, 2 W resistor
S1—SPST normally open switch
S2—102-1009-ND encoder (Digikey)
S3, S4—SPDT mini-toggle switch
T1—FT-37-77 core; 4 turns #24AWG 32 turns #26 AWG enameled wire (wind 32 turns on the whole core, then wind 4 turns on whole core)
U1—SE1206nd, 6 MHz, 8-pin DIP oscillator (Digikey)
U2—PIC16c745/JW microcontroller (Digikey)
U3—AD9851BRS DDS (Digikey)
U4—SE1714nd, 25 MHz, 14 pin DIP oscillator (Digikey)
V1—6CL6 pentode tube

chased utility package called *DriverX-USB* from Tetradyne Software, Inc. (**www.tetradyne.com**). This software allows for virtually any USB device to be controlled using Borland *C++ Builder 6.0*. Its sister package, *DriverX*, allows for low-level control of hardware devices such as PCI, serial, parallel, and interrupts. Together, these provide the programmer with the ability to write low-level code with high capability. While *DriverX-USB* is very powerful, it costs money and requires a little more learning than another approach.

Windows XP contains a generic driver for HID devices. As a result, it is possible to interface with this class of USB devices without a custom driver or driver package. A free-ware package called "HID Controller" simplifies this process. It puts control of HID devices directly into the Borland *C++ Builder* environment. "HID Controller" available on the Web at various locations; do a search. Using this software, the PC application was very simple to design and code.

A demo program package called "ReadWriteDemo" comes with HIDcontroller. This program was minimally changed to perform the necessary control. The compiled executable as well as all of the other files are available on the *ARRLWeb* (see Note 1).

Operation

The PC software should be started before the USB cable from the VFO is plugged into the PC. When the program begins to execute, a text file loads the previous (or initial) settings. After the program is running, the USB cable from the VFO can be plugged in and the USB VFO is ready for use. If you only have one USB device plugged in, the program will look like Fig 6.

The mouse can be used to select the band and the tuning increment. The status of the VFO is displayed on the right side of the panel. You can verify complete closed-loop communication by flipping the Spot and Lock switches and observing that the status updates on the screen.

When you first use the VFO, verify that the encoder tunes the VFO in the correct direction. If it is backwards, either invert the sign in the *C++ Builder* software or simply reverse the A and B leads from the encoder.

If you have multiple HIDs plugged in, the display will look something like Fig 7.

The mouse can be used to highlight the PIC16C745 line, so that the program uses the correct device. If the VFO is unplugged during program execution, it is possible to confuse the program. As a result, I recommend that the program be terminated prior to unplugging the VFO from the PC USB port.

Final Comments

The finished project is shown at work with its intended partner in Fig 8. A printed circuit board intentionally was *not* designed for this project. The intent here is to demonstrate that USB devices, and other digital projects for the ham shack, can be designed and built from scratch. While the USB VFO is handy, think of the antenna tuners,

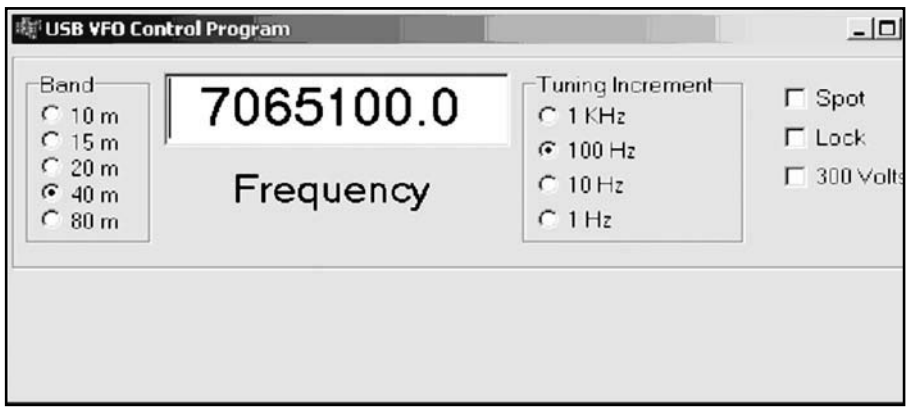


Fig 6—The interface as it appears on a computer with a single human interface device attached.

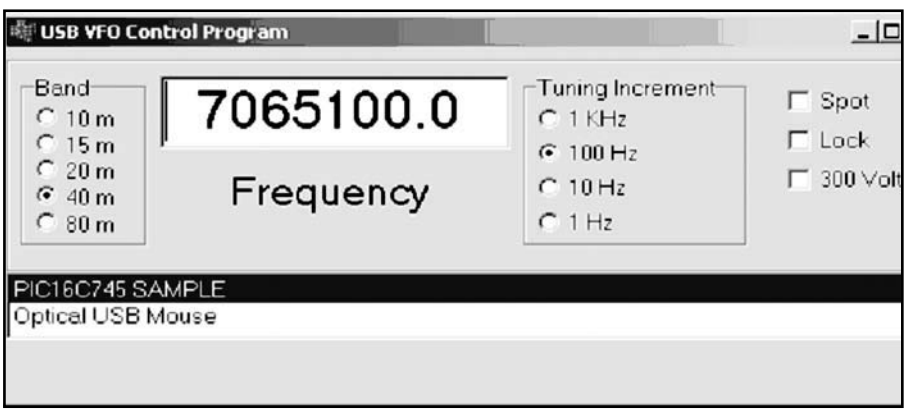


Fig 7—The interface as it appears on a computer with two human interface devices attached.

Fig 8—The VFO hard at work with the DX-60B (right). The notebook computer controlling the VFO is at left.

preselectors, RTTY interfaces and other ham-oriented USB peripherals that you could build using the same technology.

One other comment: A less powerful tube could have been used, but the 6CL6 was sitting idle in my junk box just waiting for a project. In fact, if a π-network matching and filter network is added to the plate circuit, this might make a good 1 W, or so, QRP rig by itself. With a change to a 5763 or a 6417, the rig could put out as much as 5 W.

Gary is the chief electrical engineer for DigitalGlobe Incorporated, an imaging and information company. Along with other degrees, Gary holds an MSEE from the University of Colorado at Denver. He is a member of the graduate faculty of the University of Colorado at Denver and teaches evening senior and graduate-level hardware/software classes. He is a member of IEEE and the Association of Old Crows, as well as the ARRL.

He is an Extra class operator (call WAØSPM) originally licensed in 1968 as WNØSPM (Novice) and WAØSPM (Technician class).

Although his favorite activity is working 6 m SSB, he is interested in DSP and operating digital modes on 20 meters when 6 isn't open. □□

A Fast, Simple Transverter Sequencer

Along with switching speed, this sequencer offers sensitivity (65 mW) and timing that does not vary with the strength of the keying signal.

By Tom Cefalo Jr, W1EX

Upon approaching the completion of my 10 GHz transverter project, I realized I didn't make any provisions for sequencing the RF transfer relays and power amplifier. Even though my RF power level was only about 2 W, hot switching will eventually damage the relay contacts and, in general, is bad practice. Furthermore, sequencing prevents the power amplifier from operating into an open circuit as the RF relay switches and protects the low-noise amplifier by preventing transmitted power from reaching the downconverter input.

This article describes the design of a simple, fast switching sequencer that is controlled by sensing the RF power from the transverter's associated transceiver. This makes it compatible with any transceiver and eliminates the need for any control wires. Unlike some common sequencers, the hold and delay times remain constant regardless of the detected RF voltage. The sequencer requires little operating power and correctly sequences the transfer relays and a regulated voltage to the power amplifier. In addition to this, the sequencer is not limited to low-power transverter applications but can also be extended to higher-powered systems.

51 Oak St
Winchester, MA 01890
w1ex@arrl.net

Theory of Operation

Referring to Fig 1, a small amount of energy is sampled from the main transmission through-line via a coupling capacitor. Since the value of this capacitor is very small, it does not significantly affect the main transmission line characteristic impedance. The sampled RF is converted to a dc voltage with a Schottky diode rectifier configured as a half-wave voltage doubler. The rectifier circuit is actually two half-wave voltage rectifiers. The detected voltage is then applied to the non-inverting input of the MAX942 a dual high-speed rail-to-rail comparator. It has a propagation delay of 80 ns with internal hysteresis to provide fast, clean switching in a noisy signal environment. A general-purpose switching diode in series with a Zener diode clamps the detected voltage to 5.2 V, to protect the comparator input. The switching diode's small capacitance swamps out the typically large capacitance of the Zener diode, which would otherwise increase the detected voltage rise time.

The inverting input of each comparator is biased to approximately 0.4 V by a Schottky diode used as a voltage reference. When the detected voltage on the non-inverting input of comparator A rises above the reference voltage level, the output of

the comparator switches to high (+5 V). The output of comparator A is connected to two of the 74HC32 OR gates. The output of the OR gate will be a high, unless both inputs are low. Therefore, the OR gate immediately switches to high when the comparator output also switches to a high. The output of the OR gate now forward biases the relay-driving transistors, allowing the RF transfer relays to switch. The capacitors connected in parallel with the base resistors allow the transistors to switch faster. Each transfer relay has its own driving circuit. Note that the 2N3904 transistors can safely pass 200 mA. To switch a relay requiring more current, replace the switching transistors with 2N2222As or similar (most RF relays typically draw140 mA).

The output of the B OR gate also drives an RC network, which delays the dc voltage fed to the power amplifier. The A OR gate acts as a buffer between the RC delay network and the enable input of the voltage regulator, U2. When the B OR gate switches to a high, the RC network begins to charge at a rate set by the RC time constant. When this voltage reaches a level of approximately 2.5 V (a typical transition level for HC logic) the output of the gate switches to a high level. This, in turn, enables the voltage regulator supplying dc power to the RF power amplifier. As a result, the transfer relays switch, but the dc voltage to the RF power amplifier was delayed by time constant of the RC network hence protecting the relay contacts as well as the downconverter.

A second RC network is buffered by the D OR gate is connected to the output of comparator A and buffers the voltage applied to a second RC network. This RC network allows the transfer relays and RF power amplifier to remain on or "hold" during normal speech pauses or CW character spacing. When the comparator output transitions to a high level, the RC network begins charging. As seen from Figure 1, however, this network has dual time constants. The resistor and diode connected across the RC network allow the capacitor to quickly charge because of a shorter time constant. The normally long time constant of the RC hold network is now fully charged at the rate of this faster time constant. The output of the RC hold network is connected to the noninverting input of comparator B. When this voltage exceeds the reference voltage, the output of this comparator transitions to a high level. This output is connected to the other

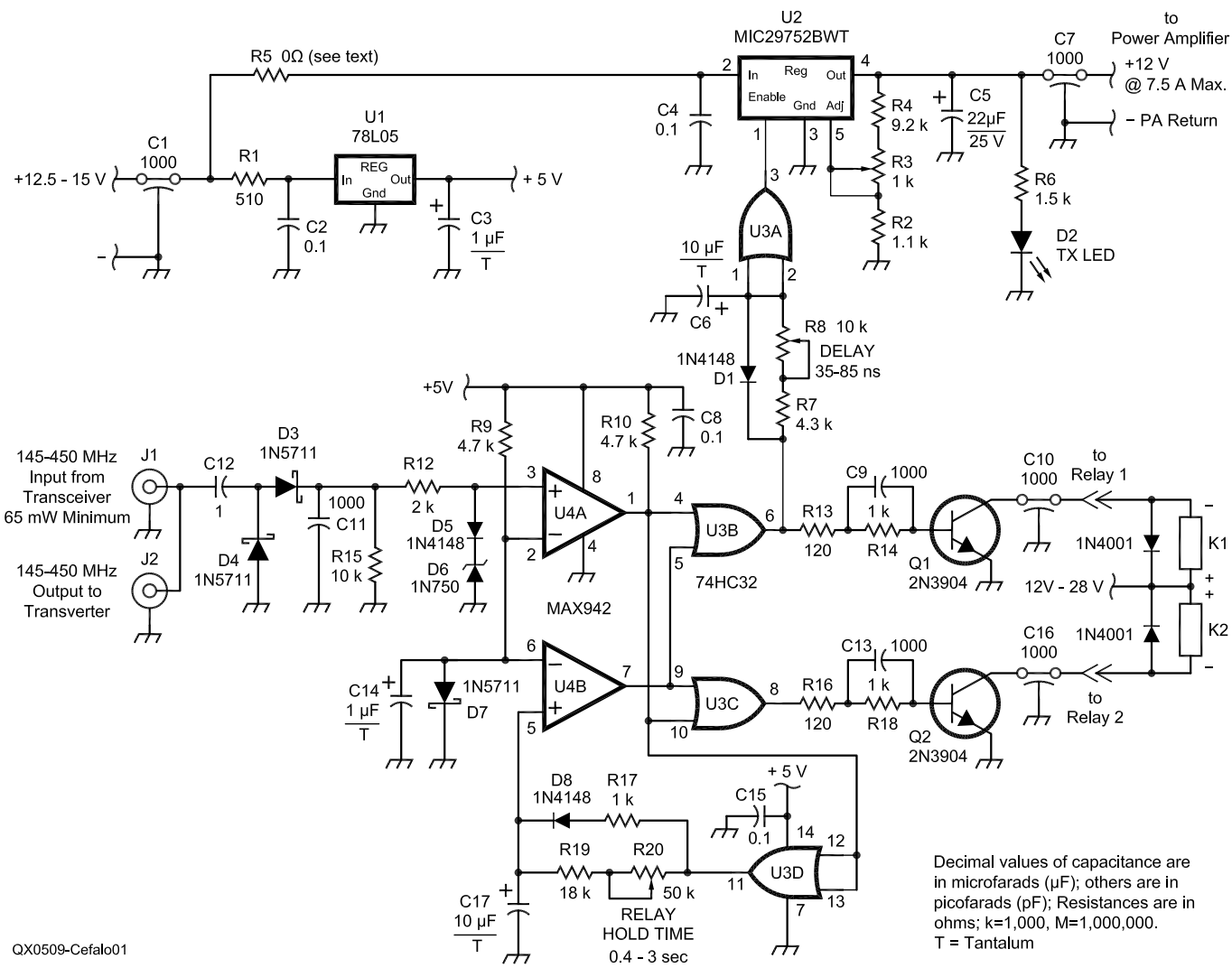


Fig 1—Sequencer schematic. Unless otherwise specified, use 1/4 W, 5%-tolerance carbon composition or film resistors. Table 1 is a bill of materials.

input pins of the relay driver OR gates.

Most sequencers are configured with the RC hold network directly after the detector diodes. The problems with this method are: first, the detected voltage becomes a function of the RC charging time, which can add a significant amount of delay between when the RF voltage is detected and when the remaining switching circuitry is triggered. Second, the hold time will vary depending on the amplitude of the detected voltage because this voltage is proportional to the RF power level. The discharge rate of the RC hold network is E_d ($e^{-t/\tau}$) where E_d is the detected voltage; consequently the hold time can change as a result of a non-constant detected voltage level. With the RC hold network connected to the output of the comparator, it's buffered and is now subject to the same voltage regardless of the RF detected voltage. With this method, the hold time is now independent of the RF detected voltage level.

As the output of comparator A varies during normal voice or CW operation, the B output will remain high because of the voltage across the RC hold network. Since the two outputs of the comparators are ORed together, the transfer relays and RF power amplifier remain on for the set hold time. The diode across this network is back biased so the longer time constant is now the dominant factor. When the transceiver is switched back to the receive mode, the detected voltage drops to zero. When the voltage charge on the RC-hold network discharges below the reference voltage level, the comparator output will switch to a low and the OR gates' output turns the transfer relays and RF power amplifier off. The diode across the RC delay network allows this network to quickly discharge and immediately disables the voltage regulator output. This ensures that the RF power amplifier will be off the next time the transceiver is keyed on.

The MIC29752 voltage regulator is a low-dropout (LDO) type. LDO voltage regulators maintain regulation with very small voltage differential between the input and output. This differential is referred to as the "dropout" voltage. Conventional linear regulators require the input voltage be a minimum of 3 V greater than the regulated output. The dropout voltage for the MIC29752 is 400 mV at 7 A and ≤100 mV at less than 1 A. The advantages of LDOs are that they are ideal for battery operation and that the regulator power dissipation is greatly reduced (Pd=($V_{dropout}$) (I)). The regulator should have generous heat sinking, depending on your application. As seen from Fig 1, a series resistor (R5) is included in the event the system is powered from a source greater than 15 V. This resistor drops the voltage applied to the regulator, which distributes the heat between the resistor and regulator. Notice that the input voltage for this regulator is limited to ≤ 26 V.

Circuit Simulation

The sequencer was first analyzed using *SwitcherCADIII*, which is a *SPICE* simulation program from Linear Technology.[1] The sequencer operation was simulated using a 145 MHz, 4 W signal. The time constants were set for a delay of 30 ms and a hold time of 1 s. Referring to Fig 2, when the detected voltage $V_{(vdet)}$, rises to 0.4 V, the output of the comparator $V_{(vcomp)}$ transitions to a high state 80 ns later. The time delay from when the comparator transitions to a

[1]Notes appear on page 28.

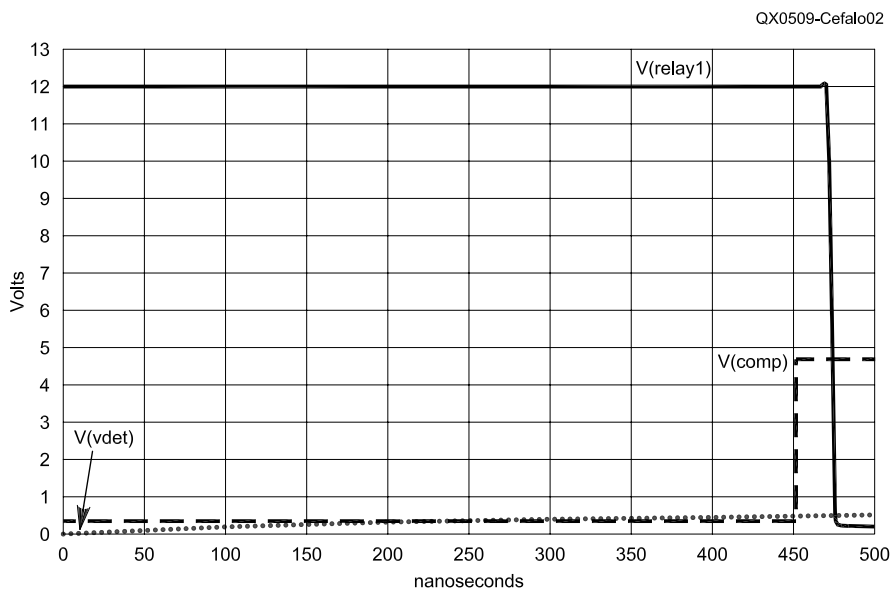


Fig 2—Sequencer turn-on time and switching characteristics.

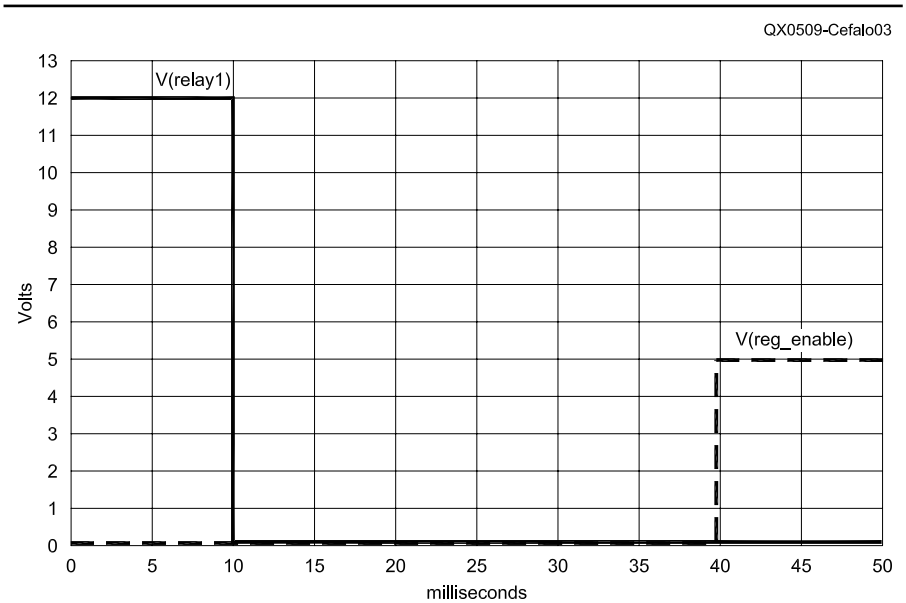


Fig 3—Sequencer RF power amplifier delay time.

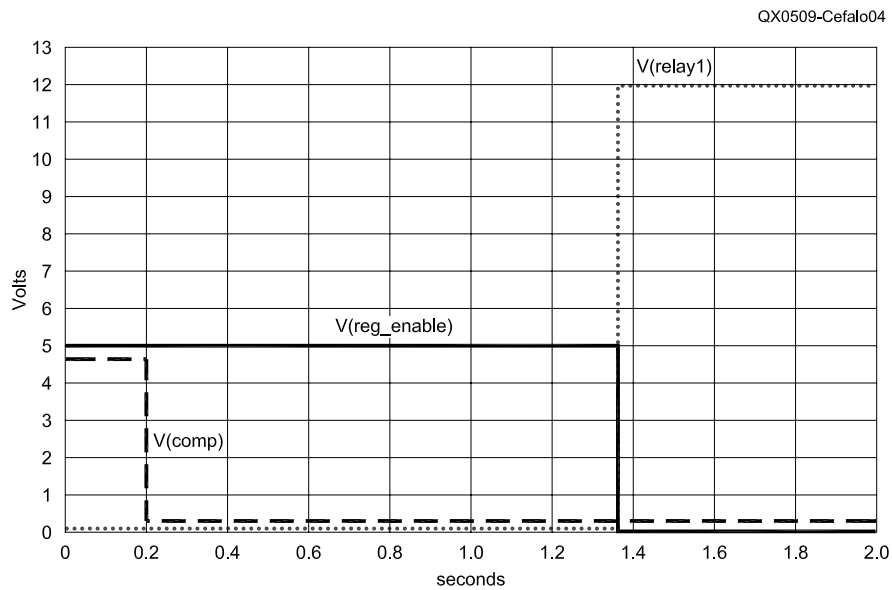


Fig 4—Sequencer turn-off time and switching characteristics.

Table 1—Bill of Materials

Item#	*Qty*	*Reference*	*Designator Value*
1	3	C9, C11, C13	1000 pF / 50 V ceramic.
2	4	C2, C4, C8, C15	0.1 µF / 50 V ceramic.
3	2	C14, C3	1 µF / 16 V tantalum.
4	1	C5	22 µF / 25 V tantalum.
5	2	C17, C6	10 µF / 16 V tantalum.
6	1	C12	1 pF / 300 V silver-mica.
7	4	C1, C7, C10, C16	1000 pF #6-32 feedthrough capacitor.
8	3	D1, D5, D8	1N4148 general purpose switching diode.
9	1	D2	LED standard type.
10	3	D3, D4, D7	1N5711 or HP5082-2800 Schottky diode.
11	1	D6	1N750 4.7 V / 500 mW Zener diode.
12	2	J1, J2	SO-239 for VHF, N connector for UHF.
13	2	Q1, Q2	2N3904 NPN general purpose transistor.
14	1	R1	510 Ω 1/4 W.
15	1	R2	1.1 kΩ 1/4 W.
16	4	R3, R14, R17, R18	1 kΩ 1/4 W.
17	1	R4	9.2 kΩ 1/4 W.
18	1	R5	0 Ω (see text).
19	1	R6	1.5 kΩ 1/4 W.
20	1	R7	4.3 kΩ 1/4 W.
21	1	R8	10 kΩ potentiometer.
22	1	R15	10 kΩ 1/4 W.
23	2	R9, R10	4.7 kΩ 1/4 W.
24	1	R11	100 Ω 1/4 W.
25	1	R12	2 kΩ 1/4 W.
26	2	R13, R16	120 Ω 1/4 W.
27	1	R19	18 kΩ 1/4 W.
28	1	R20	50 kΩ potentiometer.
29	1	U1	78L05 TO-92 5 V voltage regulator.
30	1	U2	MIC29752BWT TO-247-5 Micrel adjustable voltage regulator.
31	2	U3	74HC32 quad high-speed CMOS OR gate.
32	1	U4	MAX942 Maxim dual high-speed comparator.

high level to when the RF relay, $V_{(vrelay1)}$ switches is 25 ns (propagation delay of the gate and transistor). Therefore, it has taken approximately 480 ns from the time the transceiver is keyed to when the RF transfer relays are switched on.

Fig 3 shows the RF power amplifier delay. The time difference between the point where the RF relay turns on and the voltage regulator (V_{vreg_enable}) supplies dc power to the RF power amplifier, is 30 ms.

Fig 4 shows the turn-off characteristics of the sequencer. As the graph shows, when the comparator output transitions to a low state—for example during normal pauses in speech—the RF power amplifier and relay will remain on during this (up to 1 s) period. Finally, when the transceiver is keyed off and the 1 s-hold time constant has elapsed, both the RF power amplifier and relay turn off simultaneously.

Operation

The delay time is adjustable from15 to 85 ms, and the hold time is adjustable from 0.4 to 3 s. The switching time for a good quality RF relay is typically 15 to 25 ms. The time constants can be altered by proportionately changing R7 or R18. An LED is included to indicate when the system is transmitting. Also, notice the diodes connected in parallel with the relay coils, which squelch any inductive voltage spikes generated by the coils when the relays switch off. Most RF relays have this diode built-in internally, but it's good practice to install an additional diode—just in case.

The voltage regulator is adjustable from 12 to 13 V. This allows for any voltage drop due to IR losses in your system. The sequencer requires a nominal dc voltage of 12.5 to15 V. The sequencer circuit current requirements are 10 mA at idle and 20 mA with two RF relays, each operating at 140 mA from a separate dc supply. The sequencer has a sensitivity of 65 mW, which makes it very useful for handheld transceivers. It is usable from 144 to 450 MHz with a typical SWR of 1.06:1. The sequencer was also tested at a power level of 160 W, which makes it useful for applications other than low-power transverters.

Notes

[1] *SwitcherCADIII* is a freeware program distributed by Linear Technology. It can be downloaded from their Web site at **www.linear.com**. Look for it under "freeware programs" in the "Design Support" section.

[2] Components are available through Digi-Key or Mouser Electronics.

A Better Antenna-Tuner Balun

Which balun to use? The hybrid balun promises advantages over both voltage and current baluns.

By Andrew Roos, ZS1AN

Baluns that are situated between an antenna tuning unit and a non-resonant antenna may be subjected to high-impedance, highly reactive or unbalanced loads that can prevent the balun from functioning effectively.

In this article, I apply the analytical model of the transmission-line transformer developed by Roy Lewallen[1] to analyze the performance of the 1:1 current balun and the 4:1 voltage balun in this application. The analysis shows that the current balun operates effectively only for small load impedances, while the voltage balun is effective only if the load impedance is well balanced with respect to ground.

I then introduce a new design: the "hybrid" balun, which overcomes these limitations of the voltage and current baluns. It can operate with much higher load impedances than can current baluns and with unbalanced load impedances that voltage baluns could not drive effectively. The article concludes by describing a simple test I conducted to confirm that the hybrid balun operates as predicted.

[1]Notes appear on page 34.

PO Box 350, Newlands
7725, South Africa
zs1an@qsl.net

Lewallen's Model of a 1:1 Transmission-Line Transformer

Lewallen models the 1:1 transmission-line transformer (TLT) as an ideal 1:1 transformer with a "winding impedance," $\boldsymbol{Z}_W$, in parallel with one of the windings,[2] as shown in Fig 1. Although $\boldsymbol{Z}_W$ is shown on only one of the transformer windings, its effects are felt equally on both windings due to the coupling action of the "ideal transformer," which can be expressed by two rules:

1. The currents in the two windings of the "ideal transformer" are equal and opposite.
2. $\boldsymbol{V}_{AC} = \boldsymbol{V}_{BD} = (\boldsymbol{I}_1 - \mathbf{I}_2)\, \boldsymbol{Z}_W$

The model assumes that the length of the transmission line used to construct the TLT is short in terms of wavelength, so transmission line effects can be ignored. Although this assumption holds for the lower part of the HF region, some transmission-line effects do come into play at higher frequencies.

The use of this model does not imply that TLTs operate like conventional transformers. The ideal

1:1 transformer is simply a modeling convenience, and the resulting model applies to any device that exhibits common-mode impedance, irrespective of its operating principle.

The 1:1 Current Balun

Fig 2 shows the schematic of Guanella's 1:1 "current" balun. $\boldsymbol{Z}_1$ and $\boldsymbol{Z}_2$ represent the load. The junction between $\boldsymbol{Z}_1$ and $\boldsymbol{Z}_2$ is grounded to represent the (typically capacitive) coupling of the antenna system to ground. The load is balanced with respect to ground when $\boldsymbol{Z}_1 = \boldsymbol{Z}_2$.

The principle limitation of this balun is that it can only maintain balance between the currents flowing into the load, I_1 and I_2, for relatively low load impedances. The ability to maintain the correct current balance in the load is of course the reason for using a balun in the first place, so the extent to which a balun maintains this balance is the primary measure of its effectiveness. In the case of the 1:1 current balun, Lewallen shows that

$$\boldsymbol{I}_1 / \boldsymbol{I}_2 = (\boldsymbol{Z}_2 + \boldsymbol{Z}_W) / \boldsymbol{Z}_W \qquad \text{(Eq 1)}$$

Where $\boldsymbol{Z}_W$ is the "winding impedance" (common-mode impedance) of the balun. We can use this to calculate the measure of imbalance proposed by Witt[3] as:

$$\begin{aligned}\text{IMB} &= 2\ |(\boldsymbol{I}_1 - \boldsymbol{I}_2) / (\boldsymbol{I}_1 + \boldsymbol{I}_2)| \\ &= 2\ |(\boldsymbol{I}_1 - \boldsymbol{I}_1 (\boldsymbol{Z}_2 + \boldsymbol{Z}_W)/ \boldsymbol{Z}_W) / (\boldsymbol{I}_1 + \boldsymbol{I}_1 (\boldsymbol{Z}_2 + \boldsymbol{Z}_W)/ \boldsymbol{Z}_W)| \\ &= 2\ |\boldsymbol{Z}_2| / |(\boldsymbol{Z}_2 + 2\,\boldsymbol{Z}_W)| \qquad \text{(Eq 2)}\end{aligned}$$

(Note that $|\boldsymbol{X}|$ means the magnitude of complex variable $\boldsymbol{X}$.) If IMB is small, then $|\boldsymbol{Z}_W| >> \boldsymbol{Z}_2$. So as a good approximation,

$$\text{IMB} = |\boldsymbol{Z}_2| / |\boldsymbol{Z}_W| \qquad \text{(Eq 3)}$$

Although Witt does not suggest an acceptable maximum figure for IMB, 0.1 seems a reasonable value since this means that the magnitude of the common-mode (unbalanced) current flowing in the antenna is one-tenth the magnitude of the differential-mode (balanced) current. To prevent IMB from exceeding 0.1 we require that $|\boldsymbol{Z}_2| \le 0.1\ |\boldsymbol{Z}_W|$. This limits the usefulness of the current balun. Charles Rauch[4] measured the common-mode impedance of several commercially manufactured current baluns using a network analyzer and found that most had a common-mode impedance of less than 1 kΩ at 15 MHz, giving a maximum acceptable value of $|\boldsymbol{Z}_2|$ of less than 100 Ω. For balanced loads, this means that the baluns tested by Rauch would be effective only for load impedances under 200 Ω.

The 4:1 Voltage Balun

We can use the same model to analyze the performance of the Ruthroff's 4:1 voltage balun shown in Fig 3. Once again, $\boldsymbol{Z}_1$ and $\boldsymbol{Z}_2$ represent the load. According to Lewallen's model, the voltages across the two windings of the transmission-line transformer (TLT) are equal, and both equal $\boldsymbol{V}_S$, so

$$\boldsymbol{I}_1 = \boldsymbol{V}_S / \boldsymbol{Z}_1$$

and

$$\boldsymbol{I}_2 = \boldsymbol{V}_S / \boldsymbol{Z}_2$$

We can calculate Witt's measure of imbalance:

$$\begin{aligned}\text{IMB} &= 2\ |\boldsymbol{I}_1 - \boldsymbol{I}_2| / |\boldsymbol{I}_1 + \boldsymbol{I}_2| \\ &= 2\ |\boldsymbol{Z}_2 - \boldsymbol{Z}_1| / |\boldsymbol{Z}_2 + \boldsymbol{Z}_1| \qquad \text{(Eq 4)}\end{aligned}$$

The degree to which the currents are balanced depends only on the balance of the load impedances with respect to ground. In order to keep IMB ≤ 0.1, the impedances must be

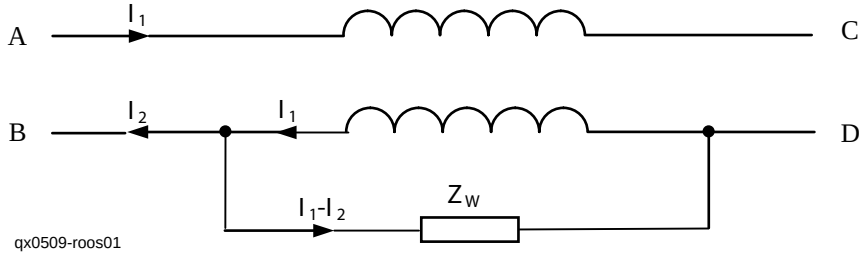


Fig 1—Lewallen's model of the 1:1 TLT.

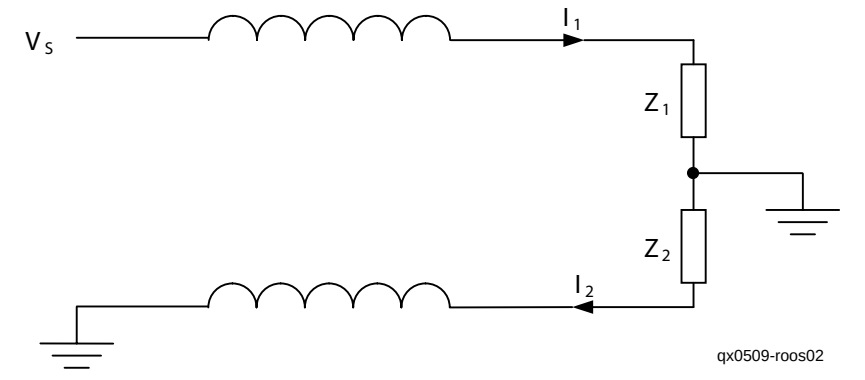


Fig 2—A schematic of Guanella's 1:1 current balun.

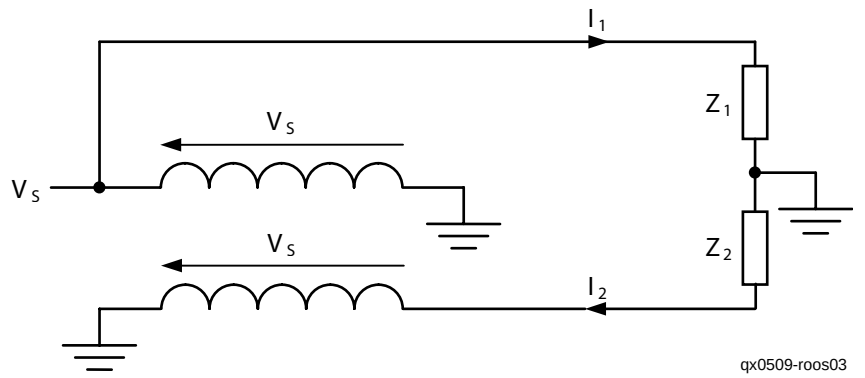


Fig 3—A schematic of Ruthroff's 4:1 voltage balun.

within about 10% of each other. Imbalance does not impose any constraint on the maximum impedance a voltage balun can drive. This is determined by efficiency considerations and is usually greater than the magnitude of the winding inductance, so the voltage balun can drive balanced loads at least five times larger than can a current balun with the same winding impedance.

Unfortunately, supposedly balanced antennas may present unbalanced loads due to the presence of nearby conductors or asymmetry in the antenna installation imposed by site constraints. Kevin Schmidt[5] describes one such case:

> "One of my antennas is a 'dipole' about 60 feet on a leg running around the outside of my 1-story house. It is fed by about 30 feet of 300 Ω TV twinlead. The legs of the 'dipole' are not straight, and it isn't symmetric about the feed point, since the shack is at a corner of the house."

He measured the differential and common-mode impedances of the antenna at 3.52 MHz using professional instruments and obtained the results shown in Fig 4.[6] If this antenna were driven by a voltage balun, IMB would be 0.22. This shows that the inherent balance of amateur antennas cannot be assumed, so antenna tuner baluns must be able to drive unbalanced loads effectively.

The Hybrid Balun

The preceding sections have shown that the current balun is limited by its inability to maintain current balance when driving all but low-impedance loads, while the voltage balun cannot maintain current balance in unbalanced loads. Since the maintenance of current balance in the load is the principle purpose of a balun, these limitations warrant a search for "a better balun." This section describes and analyses the "hybrid balun," which consists of a voltage balun followed by a common-mode choke. The circuit shown in Fig 5 uses a 4:1 voltage balun, but a 1:1 voltage balun would work equally well and the analysis is identical.

The hybrid balun requires two separate transmission-line transformers. The TLT for the voltage balun is best realized using a bifilar winding on a toroidal ferrite or powdered-iron core. The "hardy" balun designs offered by Dr Sevick[7] would be excellent in this application. I also recommend this approach for the common-mode choke. Although one could use a length of coaxial cable threaded through ferrite beads as suggested by Walt Maxwell[8] for the choke, Rauch's measurements (see Note 4) show that beaded coax chokes have a large resistive component to their common-mode impedance. Equation 13 shows that this will increase the choke's power dissipation and reduce the efficiency of the balun. The voltage balun found on many ATUs can be converted to a hybrid balun by simply adding a common-mode choke after the balun, without need for modification to the ATU or balun.

Although the common-mode choke appears identical to the 1:1 current balun, the term "common-mode choke" is preferred, as it is driven by the balanced voltage at the output of the voltage balun, instead of having one side grounded. This means the voltage across the windings is no longer the full source voltage $\boldsymbol{V}_S$, but only a portion of $\boldsymbol{V}_S$ that depends on the degree of imbalance of the load. Consequently, the common-mode choke does not suffer from the limitation of the current balun—that $|\boldsymbol{Z}_2|$ must be less than one tenth of $|\boldsymbol{Z}_W|$ in order to maintain balance and can operate effectively and efficiently at much higher impedance levels than could a current balun. In turn, the choke presents a balanced load impedance to the voltage balun preceding it, despite any imbalances in the antenna system, allowing the hybrid balun to drive unbalanced loads more effectively than could the voltage balun alone.

Let $\boldsymbol{V}_C$ be the common-mode voltage across the choke windings as shown in Fig 5. Since $\boldsymbol{V}_C$ also appears across the winding impedance $\boldsymbol{Z}_W$ in Lewallen's model, and the current flowing through the winding impedance is the common-mode current $\boldsymbol{I}_1 - \boldsymbol{I}_2$,

$$\boldsymbol{V}_C = (\boldsymbol{I}_1 - \boldsymbol{I}_2)\,\boldsymbol{Z}_W \qquad \text{(Eq 5)}$$

The voltages at the input side of the choke are $\boldsymbol{V}_S$ and $-\boldsymbol{V}_S$, so

$$\boldsymbol{I}_1 = (\boldsymbol{V}_S - \boldsymbol{V}_C) / \boldsymbol{Z}_1 \qquad \text{(Eq 6)}$$

and

$$\boldsymbol{I}_2 = (\boldsymbol{V}_S + \boldsymbol{V}_C) / \boldsymbol{Z}_2 \qquad \text{(Eq 7)}$$

Substituting Equations 6 and 7 into Equation 5 and solving for $\boldsymbol{V}_C$,

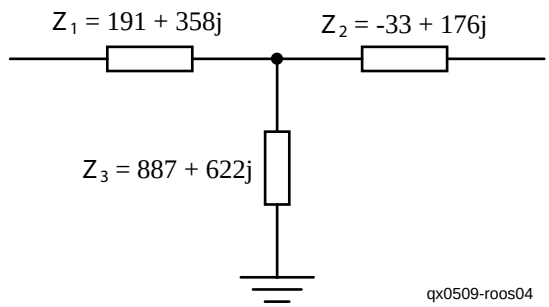


Fig 4—The impedances of Schmidt's dipole.

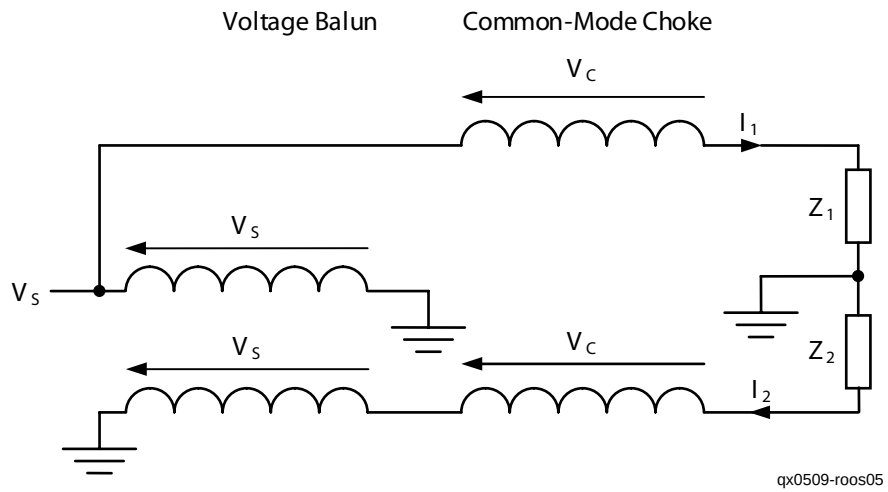


Fig 5—A schematic of the 4:1 hybrid balun.

$$\boldsymbol{V}_C = \boldsymbol{V}_S \boldsymbol{Z}_W (\boldsymbol{Z}_2 - \boldsymbol{Z}_1) / (\boldsymbol{Z}_1 \boldsymbol{Z}_2 + \boldsymbol{Z}_W \boldsymbol{Z}_1 + \boldsymbol{Z}_W \boldsymbol{Z}_2) \quad \text{(Eq 8)}$$

We can substitute this value into equations 6 and 7 to obtain

$$\boldsymbol{I}_1 = \boldsymbol{V}_S (\boldsymbol{Z}_2 + 2\,\boldsymbol{Z}_W) / (\boldsymbol{Z}_1 \boldsymbol{Z}_2 + \boldsymbol{Z}_1 \boldsymbol{Z}_W + \boldsymbol{Z}_2 \boldsymbol{Z}_W) \quad \text{(Eq 9)}$$

and

$$\boldsymbol{I}_2 = \boldsymbol{V}_S (\boldsymbol{Z}_1 + 2\,\boldsymbol{Z}_W) / (\boldsymbol{Z}_1 \boldsymbol{Z}_2 + \boldsymbol{Z}_1 \boldsymbol{Z}_W + \boldsymbol{Z}_2 \boldsymbol{Z}_W) \quad \text{(Eq 10)}$$

So Witt's measure of imbalance is

$$\begin{aligned} \text{IMB} &= 2\,|(\boldsymbol{I}_1 - \boldsymbol{I}_2) / (\boldsymbol{I}_1 + \boldsymbol{I}_2)| \\ &= 2\,|(\boldsymbol{Z}_2 - \boldsymbol{Z}_1) / (\boldsymbol{Z}_1 + \boldsymbol{Z}_2 + 4\,\boldsymbol{Z}_W)| \end{aligned} \quad \text{(Eq 11)}$$

Comparing this to Eq 4, which gives IMB for the voltage balun, we see that the additional term in the denominator reduces the imbalance by a factor of

$$\text{IMB}_V / \text{IMB}_H = |1 + 4\,\boldsymbol{Z}_W / (\boldsymbol{Z}_1 + \boldsymbol{Z}_2)| \quad \text{(Eq 12)}$$

Where IMB_V and IMB_H are the imbalance measures for the voltage balun and the hybrid balun, respectively, with the same loads.

Efficiency of the Hybrid Balun

Efficiency is as important as balance for antenna tuner baluns. The power loss in a TLT can be derived from Lewallen's model by noting that the power loss in each winding is the dot product of the voltage across the winding, $\boldsymbol{V}_W$, and the current flowing in the winding, so the total power loss in both windings is:

$$\begin{aligned} P_{LOSS} &= \boldsymbol{V}_W \cdot \boldsymbol{I}_1 + \boldsymbol{V}_W \cdot (-\boldsymbol{I}_2) \\ &= \boldsymbol{V}_W (\boldsymbol{I}_1 - \boldsymbol{I}_2) \\ &= \boldsymbol{V}_W (\boldsymbol{V}_W / \boldsymbol{Z}_W) \\ &= R\text{e}\,(\boldsymbol{V}_{W^*} (\boldsymbol{V}_W / \boldsymbol{Z}_W)^*) \\ &= R\text{e}(|\boldsymbol{V}_W|^2 / \boldsymbol{Z}_W^*) \\ &= |\boldsymbol{V}_W|^2\, R\text{e}(\boldsymbol{Z}_W) / |\boldsymbol{Z}_W|^2 \end{aligned} \quad \text{(Eq 13)}$$

Where "**X***" means the complex conjugate of "**X**" and "Re(**X**)" means the real component of "**X**." This interesting result shows that the power dissipation in the TLT is proportional to the square of the voltage across the TLT windings. So for the hybrid balun (refer to Fig 5) if the winding impedances of the two TLTs are the same, the relationship between P_C, the power dissipated in the common-mode choke and P_V, the power dissipated in the voltage balun is given by

$$P_C / P_V = |\boldsymbol{V}_C|^2 / |\boldsymbol{V}_S|^2 \quad \text{(Eq 14)}$$

Substituting the formula for $\boldsymbol{V}_C$ given in Eq 8 and canceling $|\boldsymbol{V}_S|^2$ gives

$$P_C / P_V = |\boldsymbol{Z}_W (\boldsymbol{Z}_2 - \boldsymbol{Z}_1) / (\boldsymbol{Z}_1 \boldsymbol{Z}_2 + \boldsymbol{Z}_W \boldsymbol{Z}_1 + \boldsymbol{Z}_W \boldsymbol{Z}_2)|^2 \quad \text{(Eq 15)}$$

When $|\boldsymbol{Z}_W|$ is small compared to $|\boldsymbol{Z}_1 \boldsymbol{Z}_2|$, this approaches zero, so the efficiency of the hybrid balun is the same as that of the voltage balun. When $|\boldsymbol{Z}_W|$ is large compared to $|\boldsymbol{Z}_1 \boldsymbol{Z}_2|$, it approaches

$$P_C / P_V = |(\boldsymbol{Z}_2 - \boldsymbol{Z}_1) / (\boldsymbol{Z}_1 + \boldsymbol{Z}_2)|^2 \quad \text{(Eq 16)}$$

Substituting the value of IMB for the voltage balun given in Eq 4,

$$P_C / P_V = 1/4\ \text{IMB}^2 \quad \text{(Eq 17)}$$

So when $|\boldsymbol{Z}_W|$ is large, the additional power lost in the common-mode choke is proportional to the square of the imbalance that would have existed in the load if it was driven by the voltage balun alone. This suggests the interpretation that the power dissipated by the choke depends on the amount of work the choke has to do in order to maintain the current balance in the load. If the load is perfectly balanced, the choke dissipates no power. In the worst case, when $\boldsymbol{Z}_1$ or $\boldsymbol{Z}_2$ is zero (a completely unbalanced load), IMB = 2, and the choke dissipates the same power as the voltage balun. Such an extreme case is unlikely, however, and under normal conditions (with IMB < 1) the choke dissipates less than one quarter of the power dissipated by the voltage balun, so the overall efficiency of the hybrid balun is only slightly less than that of a voltage balun driving a balanced load (which, incidentally, is the same as that of a current balun driving a balanced load, since in both cases the voltage across the TLT windings is one half of the voltage across the load).

Comparison of Balun Performance

The chart in Fig 6 shows the regions in which the different baluns will operate effectively for purely resistive loads. R_1 and R_2, the resistive (real)

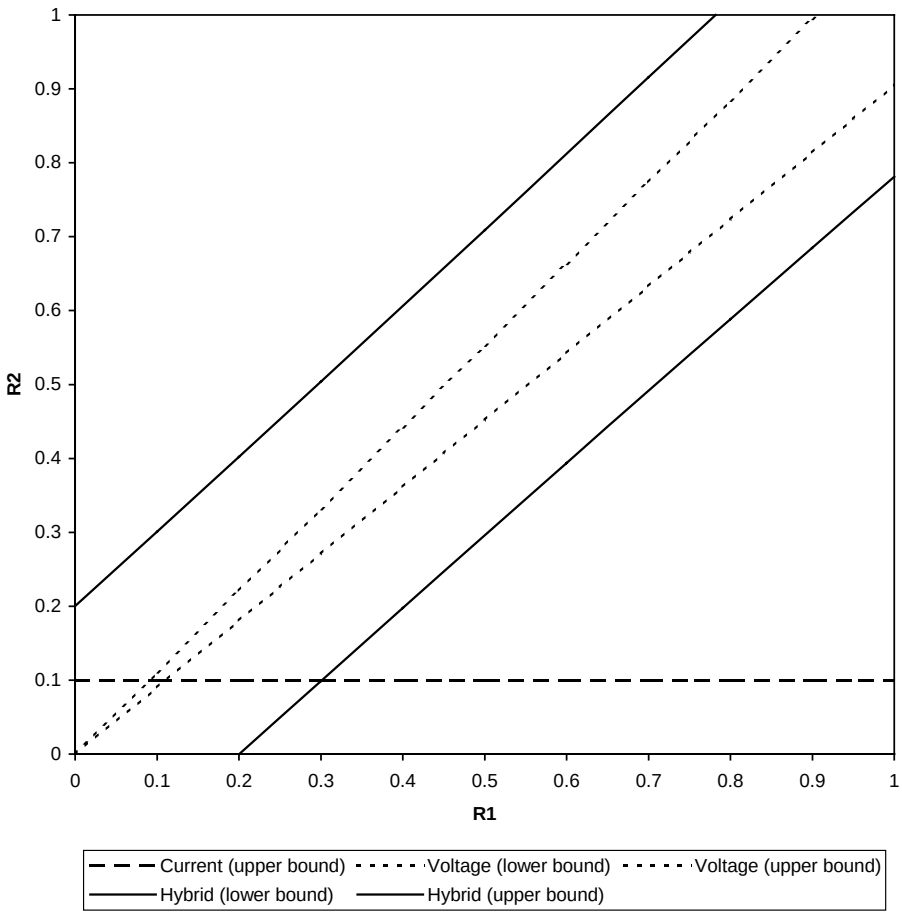


Fig 6—Effective operating regions of the current, voltage and hybrid baluns.

components of $\boldsymbol{Z}_1$ and $\boldsymbol{Z}_2$, are shown on the X and Y axes. The plot lines represent the upper and lower bounds of the regions in which each of the baluns will maintain a load imbalance (IMB) not exceeding 0.1. The chart is normalized for $|\boldsymbol{Z}_W| = 1$.

The horizontal line near the bottom of the chart is the upper bound of the current balun's operating region, showing that the current balun is effective when $R_2 < 0.1\ |\boldsymbol{Z}_W|$, irrespective of the value of R_1. (Although the current balun is able to drive high-impedance loads that are situated near the X-axis, this not terribly useful since the limiting case as $R_2 / R_1 \rightarrow 0$ is an unbalanced load being driven by an unbalanced source, so no balun is required.) The two lines that meet at the origin represent the upper and lower bounds of the region in which the voltage balun can maintain IMB ≤ 0.1. The voltage balun is effective for much larger load impedances than the current balun, but only when the load is well balanced (a perfectly balanced load would be situated on a 45° line passing through the origin). The angled lines that intersect the X and Y axes at the value 0.2 are the lower and upper bounds of the region in which the hybrid balun is effective. The hybrid balun is effective for higher load impedances than the current balun (excluding loads situated near the X axis), and for a greater degree of load asymmetry than the voltage balun.

Although the chart displays only purely resistive loads, similar results are obtained for load impedances that include both resistance and reactance. Any load that has a small resistive component combined with a large reactive component, however, may cause inefficiency and excessive heating in all balun types. If you have a load like this, you need to transform the load impedance, either by using a balanced tuner or by adjusting the length of the transmission line between the balun and the antenna.

Practical Testing

It would require more sophisticated test equipment than I have to directly measure the current balance achieved by the different types of balun. Fortunately, the input impedance of the voltage balun approximately equals the value of $\boldsymbol{Z}_1$ in parallel with $\boldsymbol{Z}_2$. This provides a good indication of load-balance quality. That indication is the extent to which the input impedance of the balun approaches the value expected for a perfectly balanced load. For the 4:1 balun, this is one quarter

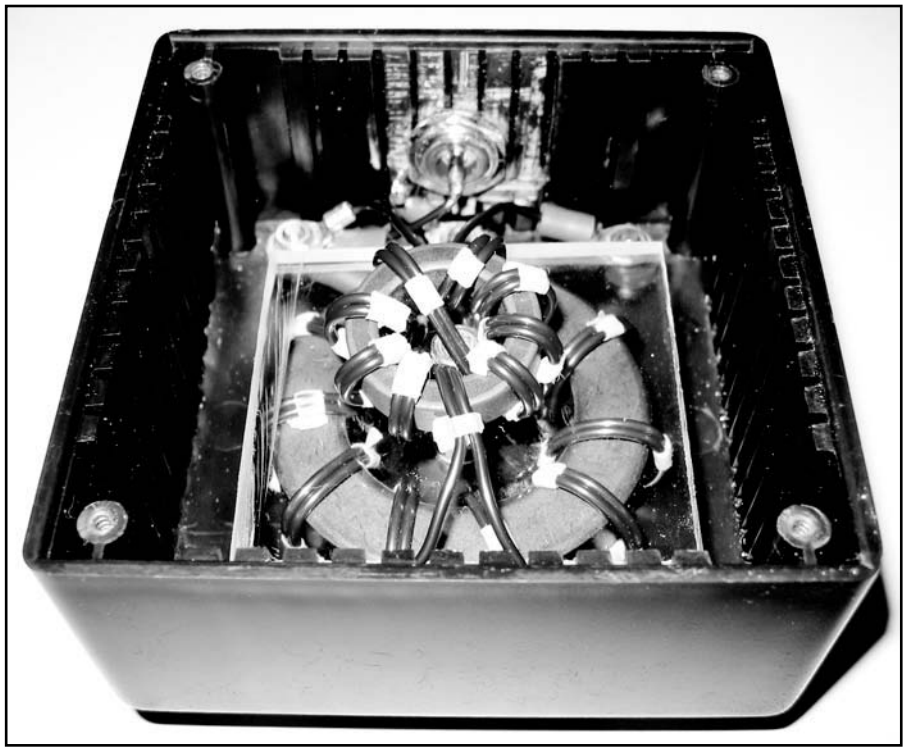


Fig 7—The prototype hybrid balun.

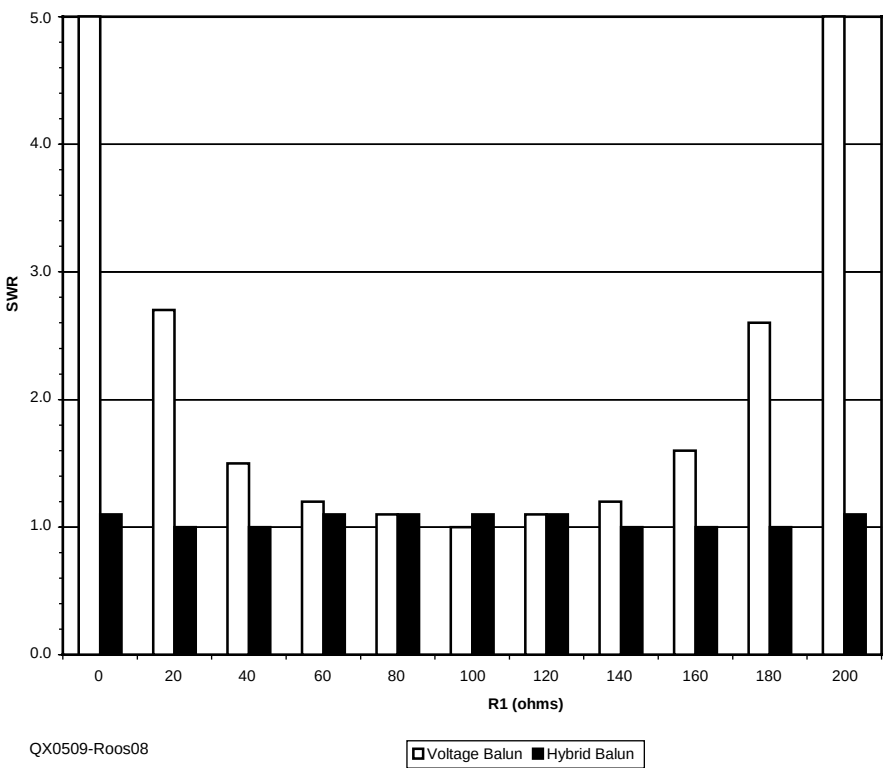


Fig 8—SWR measured at the balun input for different ground-tap positions.

of the total load impedance.

To measure this I connected 10, 20 Ω precision resistors in series to make a 200 Ω dummy load. A 12-position switch allows the load to be grounded on either side, at any of the nine junctions between resistors, or to be left floating. I constructed a voltage balun using 10 bifilar turns of #14 AWG Thermoleze enameled copper wire on an Amidon[9] FT-240-61 toroidal core, connected it to the dummy load and measured the SWR at the input of the balun when the load was grounded at each position. I then added a common-mode choke consisting of 10 bifilar turns of #14 AWG wire with a central crossover on an FT-114-61 toroidal core to make a hybrid balun and repeated the SWR measurements. All measurements were performed at 3.50 MHz. Fig 7 shows the prototype, upside-down, with the base removed. The cores used are not an optimum or recommended design, but were simply what I had on hand. With these cores, 10 turns on each core gives a common-mode impedance of 500 Ω or higher from 3.5 MHz to 21 MHz, while 8 turns per core is recommended to cover 7 MHz to 30 MHz.

Fig 8 plots the measured SWR against the value of R_1 (the resistive component of $\boldsymbol{Z}_1$). The total load impedance $(R_1 + R_2)$ is always 200 Ω. With the voltage balun, the SWR rises rapidly as the earth tap is moved away from the center of the load, indicating that the currents flowing in the load are not balanced. With the hybrid balun, the SWR remains virtually flat, indicating that the load currents are properly balanced, irrespective of whether the load impedance is balanced with respect to ground.

At frequencies above 7 MHz, the SWR measured at the input to the hybrid balun increases, although it remains constant as the balance of the load with respect to ground is changed. This does not indicate a problem with the balun; it occurs simply because the bifilar winding of the common-mode choke acts as a transmission line with a characteristic impedance of approximately 50 Ω, which transforms the load impedance of 200 Ω to a somewhat lower resistive value, with an additional reactive component. This could be avoided by using a transmission line with a characteristic impedance of about 100 Ω to wind the voltage balun, and one with an impedance of about 200 Ω to wind the common-mode choke. Sevick suggests covering the #14 AWG Thermoleze wires with 15-mil-wall Teflon tubing to obtain a characteristic impedance close to 100 Ω. He suggests "tinned #16 AWG wire... covered with Teflon tubing and further separated by two Teflon tubes"[10] for a characteristic impedance of about 190 Ω.

Conclusion

I used Lewallen's model of the transmission-line transformer to analyze the performance of the 1:1 current balun and the 4:1 voltage balun. The analysis showed that the current balun can only maintain the proper current balance in a load when the load impedance is relatively low, while the voltage balun can only maintain balance when the load is itself well balanced with respect to ground. Since current balance is the reason for using a balun in the first place, and the baluns used in antenna tuners may be required to drive loads that do not meet these requirements, neither the current balun nor the voltage balun is perfectly suited to this application.

I introduced a new design, the "hybrid" balun, and showed that it can maintain current balance while driving a wider range of loads than either the current or the voltage balun, including high-impedance and unbalanced loads. The hybrid balun is easy to construct and may be retrofitted to many existing antenna tuners without the need for any internal modifications. Although it is not perfect, I believe that the hybrid design is "a better antenna-tuner balun."

Notes

[1]R. Lewallen, W7EL, 1995, "The 1:1 Current Balun," which is available at **eznec.com/misc/ibalun.txteznec.com/pub/ibalun.txt**. Lewallen describes it as a model of the 1:1 current balun, but it is relevant to all applications of the 1:1 TLT.

[2]$\boldsymbol{Z}_W$ is a complex variable with a real component representing resistance and an imaginary component representing reactance. Complex variables are typeset here in bold face.

[3]F. Witt, AI1H, "Evaluation of Antenna Tuners and Baluns—An Update," *QEX*, Sep/Oct 2003, p 13.

[4]C. Rauch, W8JI, "Balun Test," available at **www.w8ji.com/Baluns/balun_test.htm**.

[5,6]K. Schmidt, W9CF, "Putting a Balun and a Tuner Together," available at **fermi.la.asu.edu/w9cf/articles/balun**. Schmidt gives the measurements in terms of the common-mode, differential-mode and unbalanced impedances. I calculated $\boldsymbol{Z}_1$, $\boldsymbol{Z}_2$ and $\boldsymbol{Z}_3$ using the formulae he gives.

[7]J. Sevick, W2FMI, *Understanding, Building and Using Baluns and Ununs, CQ Communications*, 2003, chapter 9.

[8]W. Maxwell, W2DU, "Some Aspects of the Balun Problem," *QST*, Mar 1983, p 38.

[9]Amidon Associates: **www.amidoncorp.com**.

[10]J. Sevick, W2FMI, *Understanding, Building and Using Baluns and Ununs,* CQ Communications (2003), pp 59 and 81.

Andrew Roos has been a radio amateur since 2001. He has a Bachelor of Arts with Honors degree from Rhodes University and is the technical director and principal systems architect of a software development company in Cape Town, South Africa. He is a past chairman of the Cape Town Amateur Radio Center, regularly teaches classes for the Radio Amateurs's examination, and recently initiated a project to introduce Amateur Radio to school children from underprivileged communities. ☐☐

Help for Oscillator Failure in the HP8640B

With a signal generator as complex as the HP8640B, there are many possibilities for failure. This is QEX*'s second article on fixing a common failure mode.*

By John H. Klingelhoeffer, WB4LNM

Losing the service of a piece of critical test equipment can be a disappointment and a discouragement to any experimenter. This is especially true when it's finally found to be a part costing pennies that is sidelining a several thousand dollar instrument. In May of 2004, that happened to me when one of my HP8640B generators suddenly and without warning lost all RF output. Due to family and work schedules, I couldn't return to troubleshooting the generator immediately. During that pause, I received the Jul/Aug 2004 issue of *QEX*,[1] with an article from Markus Hansen, VE7CA, that eloquently described his ordeal in troubleshooting and fixing output failure in his HP8640B. *Voila!* I thought. I'll just follow his easy steps and get my now 42 pound doorstop back up and running in no time. That, however, wasn't to be the case.

Fast-forward until February of 2005, when again a pause in those continuing family and work activities left me a day to try Markus' magic. I followed his well-documented path in *QEX*, only to find out that it wasn't a failure in an output RF amplifier that was causing problems in my particular case. After a bit of probing, I found that there wasn't any signal going into the output amplifiers. I could now see that this was going to call for returning to the schematics and block diagrams to track this down. Fortunately, the manual for these vintage generators is available at several sites on the Internet.[2] After reviewing the manual and doing a bit of probing, it was clear that there was no RF signal coming from the master oscillator (referred to as the "A3" assembly). All of the supply voltages going to the unit were spot on. A search of Internet resources led me to only a few e-mail reflector threads discussing this problem. The two most prevalent suggestions given were to disassemble the cavity oscillator (carefully!) to wipe out excessive silicone grease, and also baking the

1500 Kingsway Dr
Gambrills, MD 21054

[1]Notes appear on page 40.

oscillator at a very low temperature in the oven to remove moisture. Both of these were performed as suggested, but neither resulted in any output at all from the oscillator. Back to the manual and its schematics I went.

I was puzzled by the schematics showing no more for oscillator components than a single NPN transistor, internal cavity whereabouts unknown. Surely there must be some additional components to make this thing work. From the effort I'd expended in disassembling the cavity for cleaning, I knew that I'd have to be very careful in breaking down the assembly any further to find the components. About this time, I exchanged e-mail with Paul Alexander, WB9IPA, who also had experienced an HP8640B oscillator failure. In part of one discussion, Paul mentioned that the transistor was externally accessible on the oscillator for easy and quick replacement. I went back to the bench and soon found that this transistor is hidden in the open beneath a $^3/_8$ inch hex cap on the bottom of the A3 oscillator assembly (Fig 1). Removing that cap allowed me to easily lift out the TO-72 metal-cased oscillator transistor. Checking the voltages at the transistor socket that were supposed to be ±20 V showed immediately that there was no –20 V source. Finally I'd found a definitive problem to fix. But, there was a conundrum. I had –20 V on the outside of the A3 RF oscillator assembly, but none at the oscillator transistor socket, and the schematics from the Internet showed no components between those two points.

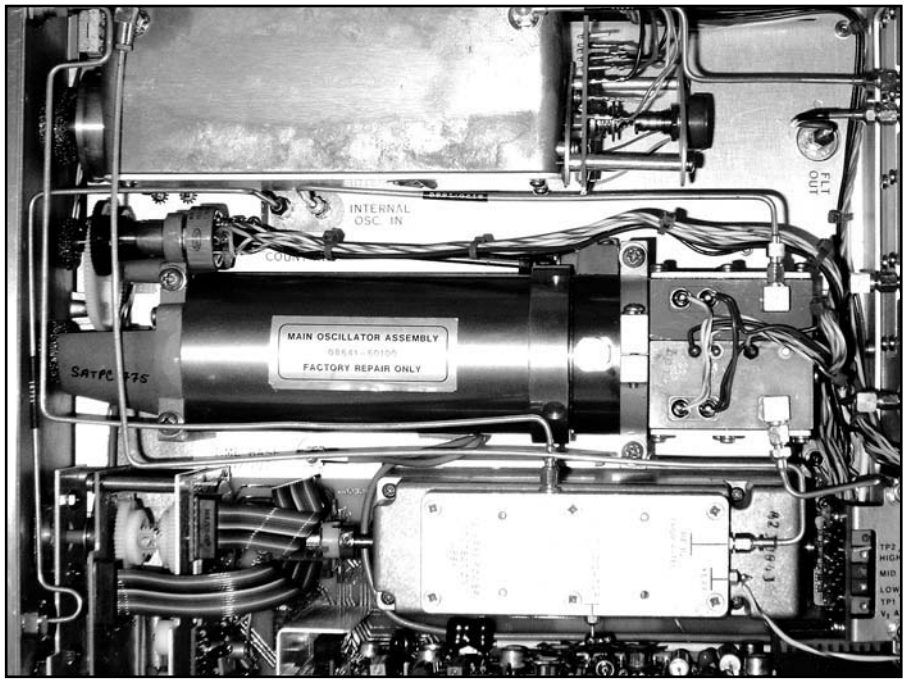


Fig 1—View of the A3 assembly to identify the position of transistor Q1.

Locating the Problem

Knowing now that I had to find where the electrons weren't flowing inside the A3 oscillator assembly, I carefully removed the assembly, more on how to do this later, and tried to determine how to get the assembly disassembled without destroying it in the process. The oscillator assembly has ends that I'll refer to as the "cone" end and the "box" end. The cone end has the shaft that protrudes through the front panel for tuning, and the box end has a group of feedthrough filters that are used to supply power and tuning signals (Fig 2).

The box end consists of small and large shielded enclosures bolted together and onto the end of the cavity. Tracing the wires carrying the ±20 V to the oscillator located the appropriate feedthough filters on the larger box. Through trial and many errors, I

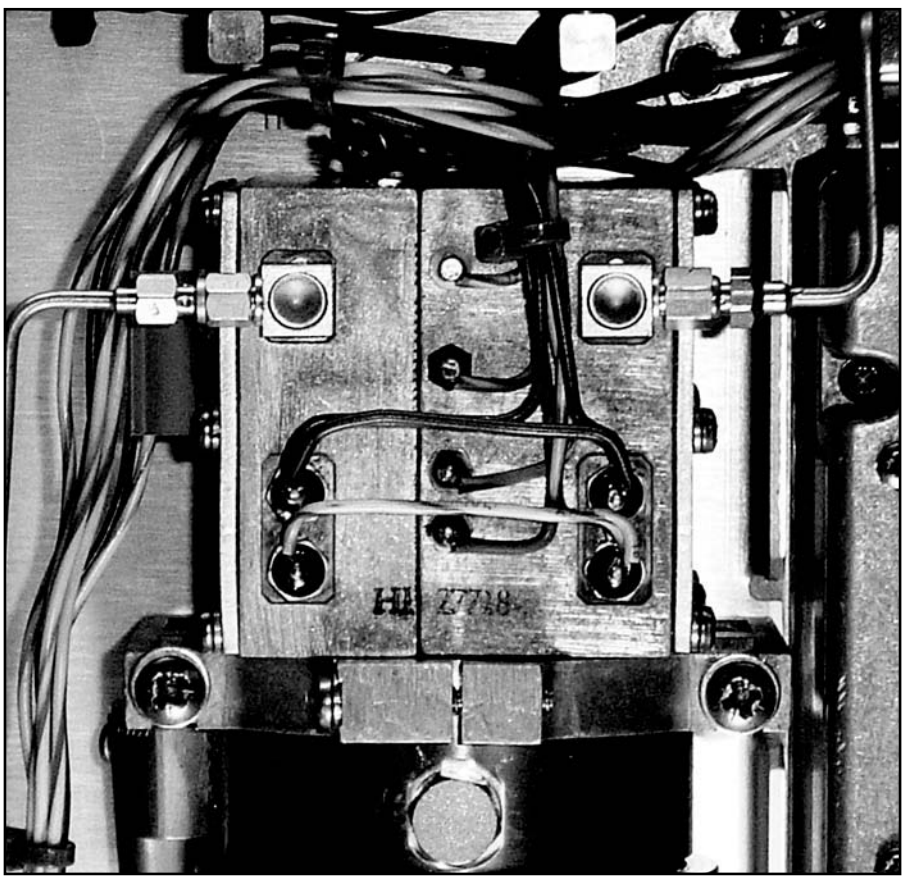


Fig 2—The box end of A3 showing feedthrough positions.

found that I could get to the internal side of the filters by carefully detaching the smaller shielded enclosure from the cavity. Interestingly, I then found three components connected to the voltage supply lines inside the base of the larger shielded enclosure that were *not* on the schematic! These turned out to be two small RF chokes and one precision resistor. A quick check with an ohmmeter showed that while both RF chokes showed electrical continuity with low resistance, the supposedly 909 Ω, 1% resistor was open and reading megohms of resistance. A quick trip to the junk box yielded a 910 Ω, 5% resistor that quickly replaced the original resistor.

By the way, there was absolutely no sign of the original resistor overheating or burning. It looked just as bright and shiny as when it was installed by Hewlett-Packard sometime in the late 1970s. Reassembling the parts and installing the A3 oscillator back in the chassis was accomplished in about 20 minutes and a power cord reattached. As soon as power was applied, the numeric LEDs showed an expected frequency in the UHF range (more on the importance of this indicator later). A check with a couple of my receivers and an external frequency counter showed that I assuredly did now have RF power output from the HP8640B.

Checking Your HP8640B for this Problem

In the preceding paragraphs, I've skipped over some of the more detailed points of the operation of the HP8640B that helped focus in on this problem. In looking back at this as an after-action discussion, if you follow the steps I'm about to discuss, you can save yourself a lot of time and trouble. If I had to do this again and the problem was the same one, it could probably be confirmed and fixed in the period of an hour (assuming one has some common hand tools and a 910 Ω resistor around the shack).

Let's start with the basics. I was using my generator and the output failed during use. At the same time as the RF output failed, the front panel digital frequency display also changed from reading the output frequency to displaying all zeros. This is a very important clue in confirming it is this problem or a very similar one (perhaps one of the RF chokes discussed earlier failing). The HP8640B has one master oscillator that simultaneously feeds two completely independent signal paths. One signal path goes to the digital frequency counter that displays the output frequency on the generator front panel. The other signal path goes to the amplifiers, modulators, and attenuators that route the RF signal to the front panel N connector on the lower right front panel. Other than the power supplies, the oscillator is the *only* common point between these circuits. Thus, if the power output fails *and* the display suddenly reads all zeros, assuming your power supply is still working properly *and* the RF ON-OFF switch is set to the ON position, the master oscillator circuit is the problem.

The first action to be taken in this troubleshooting is to confirm that you have no RF output and the display reads all zeros. This can obviously be done from the outside of the generator (and you probably wouldn't be reading this article unless one of those things was evident). Given that you have no RF output with the RF ON-OFF switch set to ON and all zeros showing on the display, you are going to need to go inside the box to do some additional checking. This would be an excellent time to turn the HP8640B POWER switch to the OFF position *and* unplug the mains line cord from the rear panel of the unit. Safety first! Those steps completed, remove the top and bottom panels from the instrument by removing four Phillips flat-head machine screws from each panel and sliding the panels to the rear of the instrument and off. The unit can then have the line cord reattached and the power turned back on. Be careful from this point on because there are lethal voltages exposed in a few areas of the chassis. The power supply voltages can be checked with a voltmeter and with the information given in the service documentation. Given that all of these voltages are nominal, we can continue with the troubleshooting.

Turn the power back OFF, and remove the line cord again for safety. Flip the chassis over so that the bottom of the chassis is now in view and facing upwards, and the front panel of the generator is facing towards you. Find the master oscillator assembly A3 on the bottom of the chassis just to the left of center (Fig 3). It will be located immediately behind the main tuning knob, and connected to it. Looking at this generally cylindrical

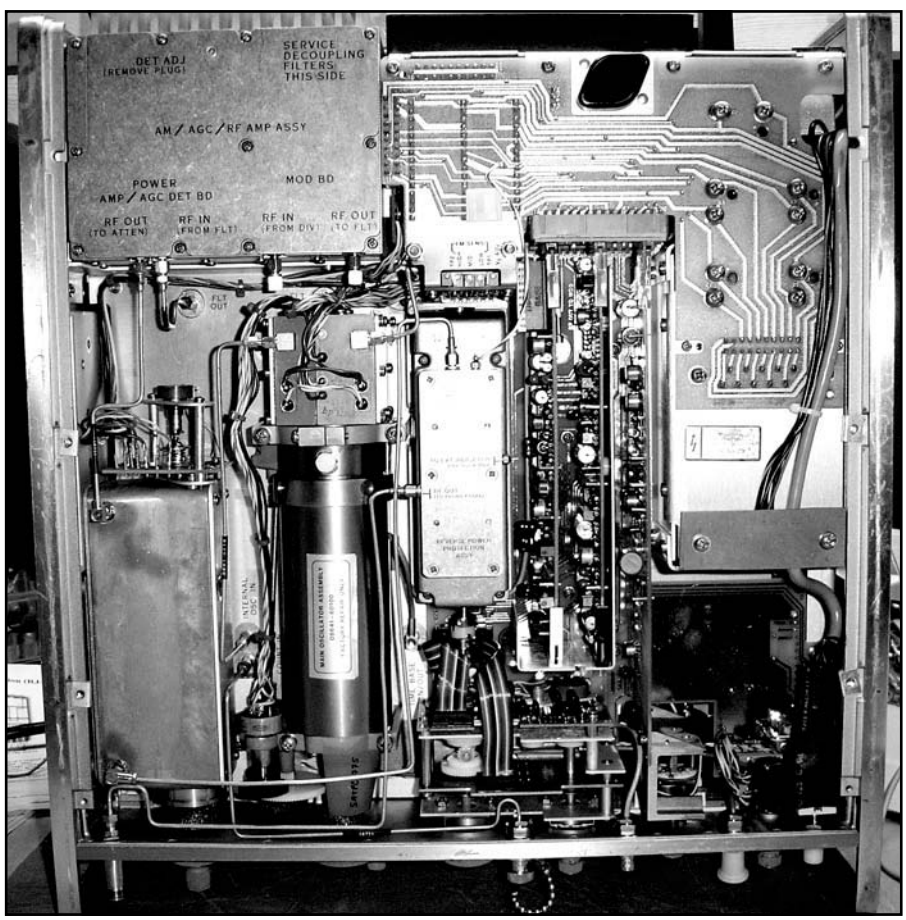


Fig 3—Chassis bottom showing the position of oscillator assembly A3.

metal object, you will find a hex head cap approximately 2/3 of the way to the box end of the assembly. Twist this cap counterclockwise to remove it. It may take a small crescent wrench to remove this if it is tight. Be careful as you remove the cap that you do not drop the small piece of wire mesh that is located in the underside of the cap. It is there to exert pressure on the transistor to keep it mechanically stable and grounded. With the cap off, you should see the top of a TO-72 metal transistor case "can," including a small reference tab at the base of the can. Take a pencil, or better yet a fine tip permanent felt tip marker such as a fine point "Sharpie," for example, and make a mark on the A3 cavity just adjacent to the transistor recess, aligning the mark with the tab on the transistor. This will assist you in replacing the transistor properly in its socket. Remove the transistor carefully by lifting it out of the recess with a pair of tweezers or with a pair of needle nose pliers with a light touch. Place the transistor somewhere safe where it won't be damaged, perhaps gently into a piece of conductive black foam.

You should at this point be looking into the transistor recess on assembly A3 and see two small sockets where the transistor was plugged in on a recessed printed wiring board inside the cavity. It's now time to plug that mains power cord back into the back panel and turn the unit back on. Remember, as soon as you reattach the power cord, there are lethal voltages within the unit. Now you are ready for the somewhat critical measurement. Set up your voltmeter to a range that can measure up to 25-30 V dc. On the "box" end of the A3 assembly, you will see two small feedthrough filters in the center of the face of the box (Fig 4). My particular unit had a wire with purple insulation going to the –20 V feedthrough and a white insulated wire with red and brown tracer stripes going to the +20 V feedthrough terminal. Using your voltmeter, check the voltage on each of these feedthroughs to ensure you have the +20 V dc on one and –20 V dc on the other, both referenced to chassis ground. Again, using your voltmeter, ground one probe to the chassis and measure the voltage at each transistor pin socket that you just exposed by removing the transistor. One socket pin should be +20 V dc and the other should be –20 V dc. On mine, I had the +20 V, but no –20 V dc. If you have one voltage, but not the other, your generator has the same or a very similar failure to mine. (See the sidebar if you find the correct voltages on the transistor socket pins.) Turn the power OFF at this point and remove the mains power cord. Using that reference mark you placed on the cavity aligned with the transistor case-tab, carefully replace the transistor into its socket seating it securely. Then replace the hex cap, screwing it tightly back in place using only your fingers (don't use a wrench to tighten this).

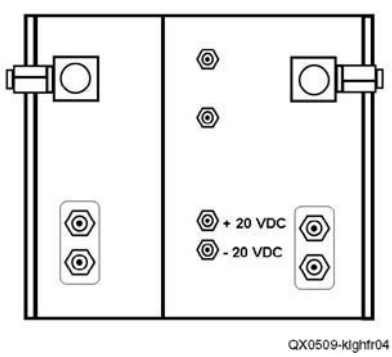


Fig 4—View of the box end of A3 identifying ±20 V feedthrough filters.

On to the Fix

You will next be finding and fixing the failed component. This involves removing the A3 oscillator assembly from the chassis, partially disassembling it, making some additional electrical measurements, and replacing one or more components. The power should be OFF at this point, and the mains cord should be disconnected from the unit. Start the assembly removal by removing the main and fine-tuning knobs from their respective front panel shafts. Each knob has two hex-head setscrews, although the sizes are different (they are also English, not metric, sizes). Next look at the bottom of the chassis and locate any semi-rigid coaxial transmission lines that cross over or pass close to the A3 oscillator assembly. The number and routing of these cables depends on options that were installed in the generator.[3] Mine had two. Carefully remove any cables that would interfere with the removal of the cavity by unscrewing the small gold plated coaxial connectors on each end of the cable. Small wrenches may help. Do not bend any of these cables while removing them, and document what they connect to by marking them with masking tape labels and perhaps a sketch of the chassis bottom components. There will also be two similar semi-rigid coaxial lines attaching to the box end of the A3 oscillator assembly. Loosen and disconnect the coaxial connectors on only the oscillator connection end of these. The cables themselves can be left attached to the other components and in the chassis. Next, find a small metal flange with a printed wiring board attached in the rear center of the chassis (Fig 5). This plate will have the words, "FM SENS" printed on it, and will also show five potentiometer adjustment screws through a slot. This plate is secured with two hex nuts. Remove the hex nuts, and gently lift the flange and printed wiring board to disconnect it from the socket below it, being careful not to damage any of the umbilical

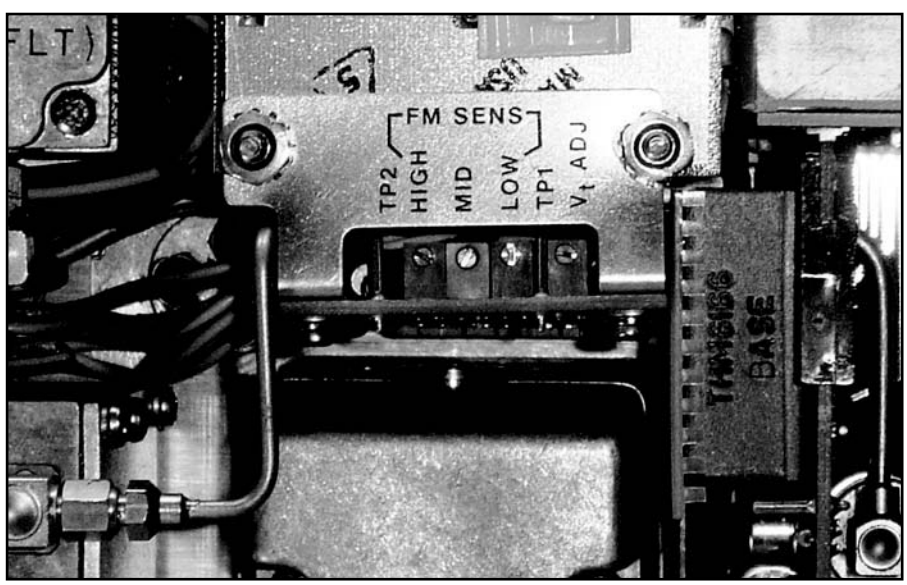


Fig 5—The umbilical cord PWB connector mounting.

cord of wires coming from the board.[4] Temporarily replace the two hex nuts on their studs so they won't get lost. Finally, look at the A3 assembly and find four round-head slotted screws, two on each side of the cylinder, that attach the assembly to the chassis. Remove these four screws and put them where you can find them again. You should now be able to carefully lift the entire oscillator assembly A3 out of the chassis by sliding it slightly to the rear so that the main and fine tuning shafts clear the front panel holes and the attached wiring harness clears any other chassis components.

Partial A3 Disassembly

Now you have the A3 assembly out of the unit and on a clean and unobstructed bench where you can work on it. You are going to remove part of the "box" on one end of the assembly where you can further troubleshoot component parts and replace defective ones. Inspect the box end of the A3, where you will see that the roughly cubic box has a larger and smaller section. You will be removing the smaller box, allowing you to see inside the larger box and replace components. First, remove the eight Phillips-head screws that hold the lid on the small box, being careful to catch the shielding screens that are captured under the box lid. Next, remove the two Phillips screws from the narrower end of the box that hold the box to the cavity. Lastly, look inside the box to find a single hex cap head machine screw and remove it with a right-angle Allen wrench. The box should now be loose from the assembly, and you should be able to carefully pull it away from the cavity, sliding the insulated probe on one end of the box out of the cavity. Again, you will find a small piece of screening material clamped between the two boxes that will fall out when the boxes are detached from one another. You should catch and protect this part for reassembly. This should expose the bottom end of the larger portion of the box, along with its recess with components inside (Fig 6).

Testing and Replacing Components

Inside the larger box recess (Fig 6) you will find a large and a small indentation. The oscillator power-feed components are in the smaller indentation. There are two RF chokes and one resistor. The RF chokes are banded wide silver-red-gold-red-silver[5], and the resistor is banded white-black-white-black-brown. Use your ohmmeter to measure the resistance of each RF choke and the resistor. The resistor in my unit had become defective and read millions of ohms instead of something around 909 Ω. The RF chokes should read only an ohm or two. If they read thousands or millions of ohms, they are defective as well, and must be replaced. I would suggest with these close quarters that once you decide which parts must be replaced, that you clip out the defective part(s) with a pair of sharp end cutters, and then quickly unsolder and remove any remaining wire loops with needle nose pliers. The small ferrite filters these components are soldered to can be destroyed quickly with soldering heat. Replace the component(s) that you have found defective and you are halfway there.

The rest of the fix is going back through this process in reverse and reassembling all of the pieces back into your HP8640B generator. Somewhat analogous in reverse to an electronics kit, you took it all apart, so you should be able to put it all back together! As soon as you get the A3 assembly back in and connected, you may want to reconnect the mains power cord and turn the power back on to confirm the correct operation prior to putting the top and bottom back on.

Results

If your disassembly, repair and reassembly went as well as mine did, as soon as you turn the power back on, you should have a reading on the front panel LED frequency counter digits that shows a frequency within the unit's tuning range based on the corresponding band switch position. If the display shows a reasonable number, go ahead and check the RF output of the generator in any manner that you wish with your other test equipment. You should not need to recalibrate anything because you haven't replaced anything that will adversely affect the specifications of the unit.

As I mentioned, in my repair I used a 910 Ω, 5% resistor. You are perfectly welcome to order a 909 Ω, 1% resistor if you wish to use an exact replacement part. I simply confirmed that the resistor that I used was within 1% of 909 Ω (not that I believe it is all that critical). As Markus said in his article, there are certainly other reasons why HP8540B generators fail, but I have seen several on eBay and at hamfests that are described as "no RF output" and "display shows all zeros" that lead me to believe that this is also a common failure mechanism.

It is a shame that a component costing only a few cents silences a fine piece of test equipment, but it does happen. Although these signal generators have been around for a generation or so, they are still more than

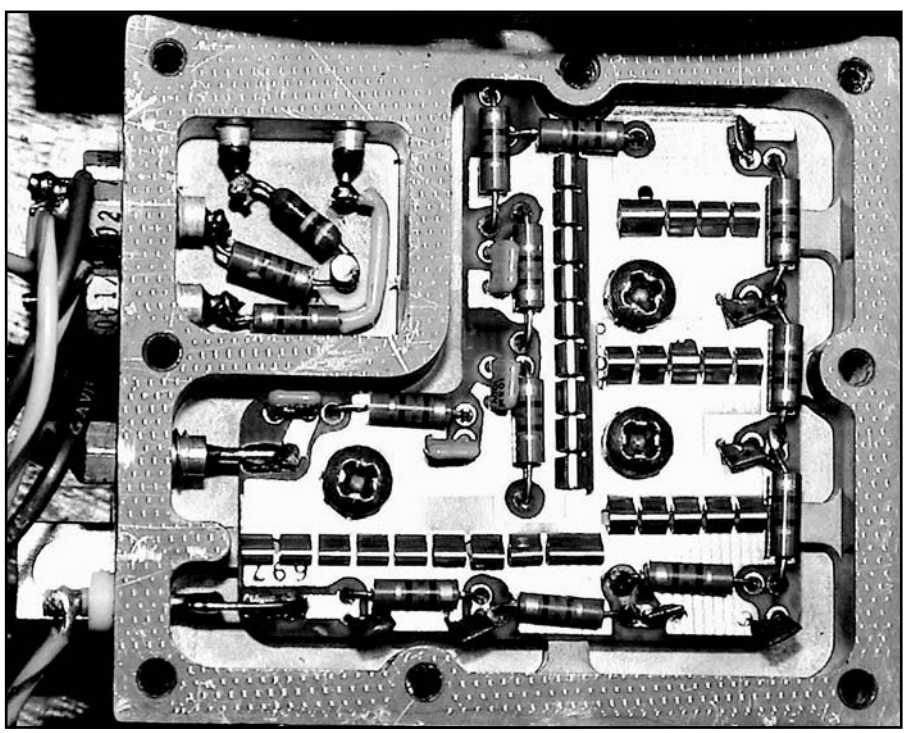


Fig 6—The recess of the shielded enclosure showing location of likely faulty components.

adequate for most amateur circuit experimentation that you would want to try. They can also be acquired for a small fraction of their original price, because most of industry has converted their test inventory to computer-controlled generators for automation reasons. This situation provides a prime opportunity for experimenters to gain a piece of top quality lab test equipment for a reasonable price.

Notes

[1]M. Hansen, VE7CA, *Help for Amplifier Failure in the HP8640B*, *QEX*, Jul/Aug 2004, pp 18-22.

[2]**bama.edebris.com/manuals/hp/8640b/**.

[3]The options for the HP8640B can be identified with these option codes:

- Option 1—Variable frequency audio oscillator,
- Option 2—Extended frequency range from 512 to 1024 MHz,
- Option 3—Reverse power protection on the output connector,
- Option 4—Specialized demodulator and other options for avionics.

[4]This board provides a jumper position for configuring the operation of the display and power to the oscillator assembly. With no jumper in position, the power to the A3 oscillator assembly is turned off each time the small RF ON-OFF switch is moved to the OFF position. The display will also change to all zeros. With the jumper in place, the oscillator stays powered at all times and the display shows the intended output frequency regardless of the position of the RF ON-OFF switch. I prefer the latter configuration, because it enhances oscillator stability, and allows the frequency lock function to remain enabled even though the RF power output is turned OFF. The disadvantage is that in that configuration, the RF output in the OFF position drops by about 40 dB instead of being completely off. In most of my experimentation the 40 dB drop is more than adequate.

[5]If the colors have survived twenty years of operation and my eyes deciphered them correctly, the value of each RF choke is 2.2 µH, ±10% tolerance. This information deciphered with the assistance of my 1975 *ARRL Radio Amateur's Handbook*.

John Klingelhoeffer, WB4LNM, is an Amateur Extra Class operator licensed and active continuously since starting with Novice call sign WN4LNM in 1968. After graduating from Auburn University with a degree in electrical engineering, he served in the US Army Signal Corps. John then spent several years with Heath Company, as a senior design engineer for VHF and UHF amateur radio products. He continued through RF engineering design and program management positions at Honeywell, Alliant Techsystems and Lockheed Martin, where he was most recently President and General Manager of COMSAT General Corporation. He is currently a Vice President with Intelsat General Corporation. He is active on VHF and UHF, continuing VHF and UHF circuit design as a hobby. He is also an advisor to the Auburn University Small Satellite Program. □□

An LMS Impedance Bridge

Come learn about LMS impedance measurement and build a unique PC sound card impedance bridge.

By Dr George R. Steber, WB9LVI

Quite some years ago, I built an impedance bridge using a relatively new (at that time) DSP microprocessor, the TMS32010. It was based on a technical paper[1] that used the LMS (least mean square) algorithm. Results of that project were mixed. It was helpful because it verified that the LMS algorithm could be used for real-time impedance measurements. But it was disheartening because it was not very accurate and could only operate at a maximum bridge frequency of 50 Hz. So I filed it away for future reference.

Over the years I've maintained an interest in impedance measurement, digital signal processing (DSP) and programming. Recently I did a novel project[2] using a PC with a sound card and some DSP techniques to implement a low-cost curve tracer (I versus V) for devices like Zeners, LEDs and transistors. During that task I sometimes thought of the old LMS bridge. Could it be implemented on a PC? After a lot of study and some serious modifications, the answer is a resounding "yes." In fact, it turned out to be much more than I had hoped for, yielding a wide range, low-cost impedance measuring system. An abbreviated article describing construction and operation of the LMS bridge has been written for *QST*.[3] Presented here are the technical details behind this unusual system, its operation and some additional practical applications. In case you don't have the *QST* article handy, I will also present some material on installation and operation of the bridge.

Nearly any PC can be used in this project, as there is no need to modify in any way. You can use one of the newer 3 GHz PCs or dust off that old 200 MHz PC that's sitting on the shelf. No need to open the cabinet either since access to the sound card stereo line jacks is all that is required, and that can usually be accomplished from a panel on the rear of the computer. Of course I am not giving out guarantees that this project will work with your system, but I will say that I have tested it with a 200 MHz Pentium Pro, a 500 MHz Pentium III and a 1.1 GHz AMD Athlon processor running *Windows 98* or *XP* with a Sound Blaster (SB) Live! sound card.

So, if you have a Pentium or AMD PC with a *Windows* compatible full-duplex sound card, you may have the basis for a very good *Windows* based impedance measuring system. All you need to do is build the simple circuit described, connect it to your computer sound card and run the program. This impedance bridge allows you to automatically measure inductors, capaci-

[1]Notes appear on page 47.

9957 N River Rd
Mequon, WI 53092
steber@execpc.com

tors, resistors, input impedances, audio transformers, negative resistances and more at a wide range of audio frequencies. It has outstanding capabilities and accuracy.

The cost of the project is less than $1 (yes, one dollar!) not counting the PC and power supplies. The circuit uses only two resistors and a dual op amp. It can be built on a solderless breadboard like I did or you can design a printed circuit board for it. In any case you will need a digital voltmeter (DVM) for calibration purposes, although even this is not absolutely required.

As usual, I am getting ahead of the story. As a professor, now retired, I am obliged to present more background and theory on this subject. I think you will find it interesting, so please try to *resist* the urge to skip to the end of the article.

Impedance

Impedance is basically the opposition to current flow. It is a more general form than resistance alone. Impedance can have a resistive part and a reactive part. For resistors, the reactive part is very small, unless they're wire wound. Inductors and capacitors have both resistive and reactive parts. The resistive part is often modeled in series with the reactance. Although the series model is used here, other models are sometimes used with parallel resistors. At a given frequency, impedance can be written in either polar (vector) or rectangular form as in (Eq 1).

$$Z = |Z| \angle \theta = R + jX \qquad \text{(Eq 1)}$$

where Z is impedance in ohms, $|Z|$ is the magnitude of Z, θ is the angle of Z, R is the real (or resistive) part of Z, and jX is the imaginary (or reactive) part of Z. The two forms in Eq 1 are related by:

$$|Z| = \sqrt{R^2 + X^2} \quad \text{and} \quad \theta = tan^{-1}\left(\frac{X}{R}\right) \qquad \text{(Eq 2)}$$

As noted above, impedance can also be written in other forms but in this article we will always model the unknown impedance as a series combination of resistance R and reactance X.

There are numerous instruments available to measure impedance including the ubiquitous ohmmeter for resistors, resistance bridges, ac bridges for capacitors and inductors, automatic LCR bridges and vector impedance meters.

Learning From The Old LMS Bridge

Since many of the ideas for the current project were derived after looking at the problems of the original LMS bridge project, we will look at it first. It is shown in Fig 1. The signals are all sampled signals, but we will not denote that at this time for sake of clarity. In this bridge, Vr and Vx are two sinusoidal voltage sources with the same radian frequency ω_0, but with different amplitudes and phase shifts. Reference voltage source Vr is of constant amplitude A and zero phase shift. However, Vx has a variable amplitude and phase shift. They can be written as follows:

$$Vr = A\sin(\omega_0 t)$$
$$Vx = B\sin(\omega_0 t + \phi) \qquad \text{(Eq 3)}$$

The parameters B and ϕ are controlled to balance the bridge. Vr and Vx are generated via D/A (digital to analog) converters from the microprocessor. Voltage $e(t)$ is read into the microprocessor with an A/D (analog to digital) converter. Other elements of the bridge are the unknown impedance Z_X and the reference resistance Rm. When the bridge is balanced (voltage $e = 0$) the unknown impedance is given by

$$Z_x = Rm\frac{B}{A}\angle\phi \text{ at the frequency } \omega_0 \qquad \text{(Eq 4)}$$

Expressing V_X in terms of in-phase and quadrature components yields:

$$V_X = B\cos\varphi\sin(\omega_0 t) + B\sin\varphi\cos(\omega_0 t) = W_1 A\sin(\omega_0 t) + W_2 A\cos(\omega_0 t) \qquad \text{(Eq 5)}$$

where $W_1 = (B/A)\cos\varphi$ and $W_2 = (B/A)\sin\varphi$ are the weights of the in-phase and quadrature components, respectively. With B and ϕ expressed in terms

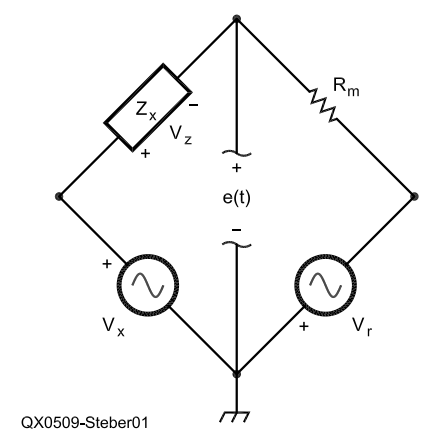


Fig 1—Old LMS bridge. Zx is the unknown and Rm is reference resistor.

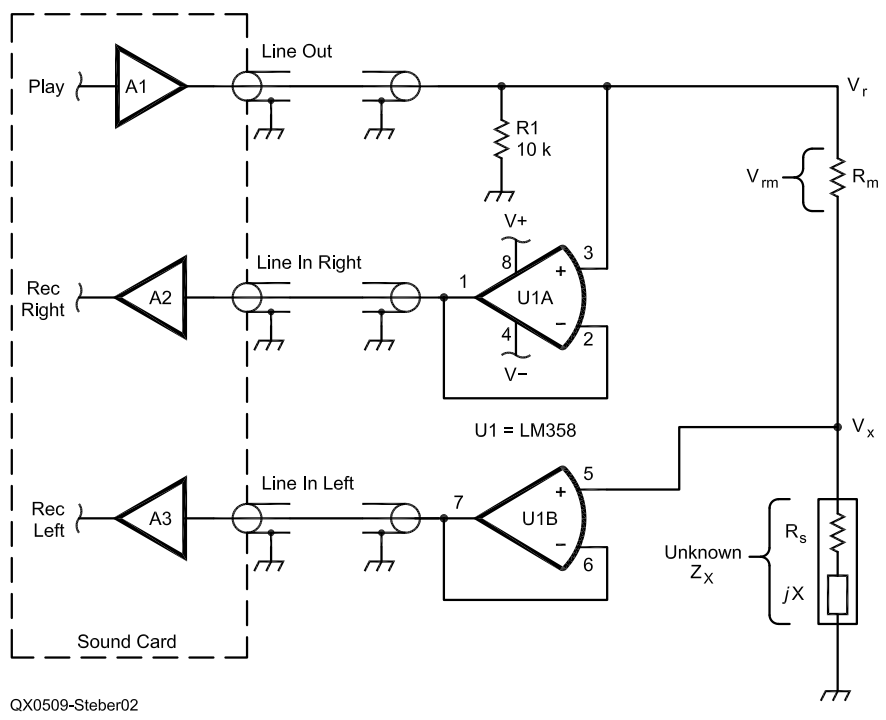


Fig 2—Impedance measuring circuit and interface to sound card. See text for more information.

of W_1 and W_2, Eq 4 can be written as:

$$Z_x = RmW_1 + jRmW_2 \text{ at balance} \quad \text{(Eq 6)}$$

The terms RmW_1 and RmW_2 are the real and imaginary parts of Z_X. To balance the bridge, one starts with initial values of W_1 and W_2 and iteratively modifies these to force e(t) to zero. One method of doing this, the LMS algorithm, requires that the error be found at each new sample and updated values of W_1 and W_2 be calculated that hopefully force e(t) to zero over time. The LMS algorithm does that well but we won't go into how it does it right now.

First we observe a few things about this bridge. On the *plus* side we see from Equation 6 that 'at balance,' the real and imaginary parts of Z_X depend only on the weights and Rm (reference resistor) and there is no requirement to know the amplitude A. That is a nice result. On the *minus* side, since the LMS algorithm requires that calculations occur at each new sample of e(t), we have only one sample period to find new values of W_1 and W_2. In addition, the sampling of e(t) and the outputs Vr and Vx must be synchronized. Note too that since the sources Vr and Vx are generated digitally they must be calculated for each new point on the sine waves. Finally, we see that the unknown, Z_X is floating above ground, which is not desirable.

Consider now a typical sound card in a PC. It has very good 16 bit A/D and D/A converters. We can easily generate the signals Vr and Vx and output them to the bridge via the sound card line outputs. Similarly, the error signal e(t) can be read with a line input. But how do we keep the inputs and outputs synchronized within one sample period? If you are familiar with *Windows* you probably know that all sound card I/O is done via buffers. And *Windows* decides when to empty and fill them. Because the input and output buffers may be long and not synchronized this argues against using the LMS algorithm, which needs to make sample to sample decisions. I wrestled with this problem, but could make no headway until I started looking at different circuit topologies.

Impedance Measuring Circuit

Consider the circuit in Fig 2. Instead of using two sound card outputs and one input, as in the old LMS bridge, it uses one output and two inputs. By intentional design, the two inputs are synchronized with each other but not necessarily to the output from the sound card. Resistor R1 is there to provide a ground reference for the sound card output. Rm is the reference resistor and Zx is the unknown, as in Equation 1. We see that the unknown impedance is now grounded. Two op-amps U1A and U1B provide isolation and buffering of the bridge voltages. They are connected as unity gain, high input-impedance, low output-impedance drivers. Vr is the sinusoidal voltage applied to the circuit via a line output of the sound card. It is fed back to the sound card input *right* channel via U1A. The voltage across the unknown Zx is buffered by U1B and fed back to the sound card input *left* channel via U1B. Shielded audio cables are suggested for connecting to the sound card.

U1 is a cheap LM358 dual op-amp or equivalent which can be powered from bipolar power supplies of 3 volts to 15 V. It is best to keep the power supply voltages low to protect the sound card line inputs in case of problems. My circuit runs at about plus and minus 3 V. A bipolar battery supply could also be made using four AAA batteries with a center tap connected to ground.

Since all measurements depend on Rm, it is critical that we know its resistance precisely. To effectuate different ranges of the instrument, we may also wish to suitably change the value of Rm. We will discuss more about this later on.

There are many ways to measure impedance with the circuit of Fig 2. All of them require that a sinusoidal signal Vr be applied to the circuit and a number of sequential samples of Vr and Vx be captured in buffers. Discussed below are several methods of doing this.

Three Measurement Method

An old method of calculating impedance, sometimes called the three-voltmeter method, can be used. Here is how it works. Measure and record the three voltages Vr, Vx and Vrm as shown on Fig 2. We see that

$$Z = \frac{V_x}{V_{rm}} Rm = Z\angle\theta \quad \text{(Eq 7)}$$

To find the magnitude of *Z,* simply take the magnitude of *V*x divided by magnitude of *V*rm and multiply by *R*m as shown below.

$$|Z| = \frac{|V_x|}{|V_{rm}|} Rm \quad \text{(Eq 8)}$$

It is a little more complicated to show, but considering the three voltages as vectors in a triangle and applying the law of cosines results in the equation for the phase angle θ as shown below.

$$\theta = 180 - cos^{-1}\left[\frac{Vr^2 - Vrm^2 - V_x^2}{2|Vrm||V_x|}\right]\left[\frac{180}{\pi}\right] \quad \text{(Eq 9)}$$

in degrees

There are other ways to calculate θ that involve multiplying the voltages and filtering, but they don't provide more benefits. A drawback to this way of finding θ is that it cannot distinguish between positive and negative angles of reactance. However, the correct θ can be found by looking at the zero crossings of $V\text{rm} = A \sin(\omega t)$ and $V\text{x} = B \sin(\omega t + \theta)$ since at $t = 0$ (and multiples of the period), $V\text{x} = B \sin\theta$ which is > 0 for inductances and < 0 for capacitances.

This method was implemented on a PC with good results over much of the measuring range. If you looked no further, this would be a an acceptable method of finding impedance. When either *V*x or *V*rm is small, however, there is a substantial error in θ. This was attributed to noise and other errors from the measurement of these voltages. More sophisticated methods can reduce these errors.

Least Squares Method

The literature is full of different error-minimization criteria but the most widely used one is the least square approximation originated by Gauss. Simply stated, the least squares principle involves selecting the function that minimizes the sum of the squared errors. A more complete discussion can be found in Reference 4. It is applicable to both continuous and discrete systems. Since we will be dealing with discrete variables for the rest of this discussion, it is appropriate to introduce them now. Fig 3 shows a adaptive linear combiner we will use to illustrate the least squares method. It consists of unit-time delays, weights and summation blocks. There are two inputs: the data samples x_k and the desired response d_k. Sample delays are represented by z^{-1} with the samples taken at points k, k–1,…, k–L, going back in time through the data samples.

The L-input samples, x_k, x_{k-1}, … x_{k-L} may be represented as a vector.

$$X_k = [x_k\, x_{k-1} \ldots x_{k-L}]^T \quad \text{(Eq 10)}$$

where T stands for transpose. So X_k is actually a column vector and the subscript k is used as a time index. Similarly, we define a weight vector

$W_k = [w_{0k}\ w_{1k} \ldots w_{Lk}]^T$ (Eq 11)

From Fig 3, we see that the output, y_k, is a linear combination of the input samples and the weights. The error signal with time index k is the difference between the desired response d_k and the output y_k and is given by:

$e_k = d_k - y_k = d_k - W^T X_k$ (dropping the subscript of W for clarity) (Eq 12)

and

$e^2_k = d^2_k + W^T X_k X^T_k W - 2d_k X^T_k W$ (Eq 13)

This is the instantaneous squared error and is the function we wish to minimize by adjustment of the weights, *W*. This rather imposing task can be attacked via two main methods, the *non-recursive* approach called the Wiener-Hopf method or the *recursive* approach of the Widrow-Hopf (LMS) method. Solutions presented in Note 4 will be used.

A closed-form solution for the Wiener-Hopf method can be written as:

$W^* = R^{-1}P$ (Eq 14)

where $R = E[X_k X^T_k]$, $P = E[d_k X_k]$, E denotes taking the expected value, and W^* is the optimal weight vector. The procedure is straightforward. We capture L samples, calculate the expected values for R and P and simply calculate W^*. Surprisingly, in practice, it works quite well.

The LMS method proceeds by first calculating the error as in Eq 12 and then applying a steepest-descent algorithm to find W_{k+1} as:

$W_{k+1} = W_k + 2\mu\varepsilon_k X_k$ (Eq 15)

where μ is a constant that controls the speed and stability of adaptation. This deceivingly simple recursive equation obtains the same solution as Weiner-Hopf, when converged.

Adaptive Impedance Bridge

Refer again to Fig 2 and notice that

$$Vx = \left(\frac{Vr - Vx}{Rm}\right)Z = \frac{Vrm}{Rm}Z \quad \text{(Eq 16)}$$

where *V*rm = *V*r–*V*x. Now, let *V*rm = $A\sin(\omega_0 t)$ and *V*x = $B\sin(\omega_0 t + \theta)$. Following the same procedure as for the old LMS bridge, *V*x can be written as:

$Vx = W_1 A\sin(\omega_0 t) + W_2 A\cos(\omega_0 t)$ (Eq 17)

To find the weights, we apply the methods described in the previous section. Fig 4 illustrates the adaptive linear combiner for this case where *V*rm is the input, *V*x is the desired signal and e_k is the error. The unknown impedance is given as:

$Z = RmW_1 + jRmW_2$ (Eq 18)

Both the Weiner-Hopf and LMS methods were implemented and simulated for comparison of accuracy and convergence using real data captured from a sound card. Many different cases of unknown impedances, whose values were precisely measured on a commercial LCR bridge, were tested. A sound card sampling frequency of 44,100 samples per second and a sinusoidal frequency of 1225 Hz was used for the test signal. Each channel was set to a capture length of 11,025 samples, which provided 0.25 seconds of data. As expected, both methods tended toward the same, and correct, solution for *Z*.

The Weiner-Hopf method provided the best (lowest) error for this number of samples in most cases. The LMS method varied in error depending on the amplitude of the signals and the value of μ. If nothing further were considered, Weiner-Hopf would be the method of choice. However, further experimentation showed that if the sampled data in the capture buffer was re-iterated several times, the LMS method improved dramatically. This, in effect, increases the number of samples. Since the LMS algorithm is so efficient, only a few milliseconds is added to the calculation. A similar reiteration for the Weiner-Hopf method does not provide additional improvement and would require a larger capture buffer. Finally, the LMS algorithm was normalized to improve the speed of convergence over a wide range of signal levels.

Both methods are very good and greatly surpass the three-measurement method described earlier. Since I wanted to keep the capture length small (in order to have about 4 measurements per second) I chose to implement the LMS method.

Sound Card Considerations

A low-distortion, low-noise full-duplex sound card is desirable. The Sound Blaster Live fills the bill nicely

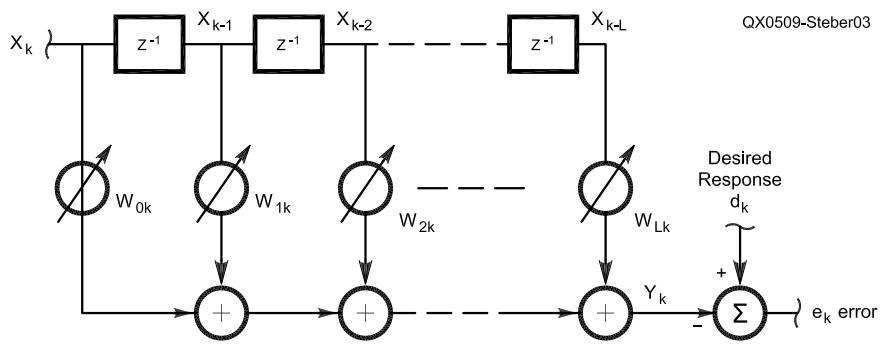


Fig 3—An adaptive linear combiner.

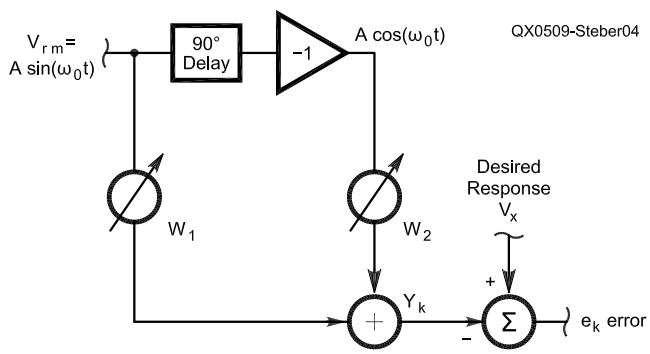


Fig 4—An adaptive combiner for the LMS bridge.

and probably many others will too. Since I cannot test them all, I will restrict my attention to this one. Referring to Fig 2, we see that A1 is the line output amplifier, and A2 and A3 are the right-and left-channel line inputs of the sound card. These amplifiers can be a source of distortion if the proper levels are not maintained.

There are two culprits here: One is excessive drive and the other is saturation. If A1 sources too much current, it will distort. I viewed *V*r on my Tektronix TDS 360 (Digital Real Time Oscilloscope with FFT) and saw a lot of second- and third-harmonic distortion when *V*r exceeded 820 mV with *R*m =10 Ω and *Z*= 0 (a short circuit). Since the full output level is 1.62 V, it needs to be attenuated. This can be done either with the sound card mixer or with the signal generator level control in my program. I chose to set the *Play* level to maximum in the mixer and set the level to 0.5 in the program, as it is easier to remember. One of the nice things about the LMS bridge is that you don't need to know the amplitude of *V*r.

The other consideration is that the line input amps A1 and A2 will saturate if the input voltage is too high. This is so regardless of the *Record* setting in the mixer. On the SB this occurs at 820 mV. Since we are using gains of "one" in the circuit, we can prevent this by adjusting *V*r as noted above. So, for my SB Live, I just set the output level to 0.82 V and both conditions are satisfied. Just in case, I have provided a real-time oscilloscope function in the bridge display, so the sine waves can be monitored for possible flat topping. If a digital voltmeter is handy, *V*x can be measured and used to calibrate the scope. This is provided in the program, but it is not required and does not affect operation of the bridge.

One other consideration is the balance of the two input channels. Since we need to calculate *V*r–*V*x precisely, the two channels must be balanced. This function is also provided in the calibration section of the program.

A note is in order about earlier Sound Blaster sound cards such as the SB16 and AWE 32, since there are so many of these still in service. Unfortunately, they do not provide true, full-duplex operation. The same may be said of SB *compatible* cards, so be wary. For example, (with the latest drivers) the SB AWE32 can simultaneously play only unsigned 8 bits and record signed16 bits. It also has a built-in amplifier that may overdrive the LMS circuit. After some extensive tweaking, I managed to get one working with this program, but the results were not as good. I advise you not to use any of these cards.

Impedance Bridge Installation and Operation

The LMS bridge software is available on the ARRLWeb site and is zipped for fast downloading. Unzip it to a new folder and you are ready to go. Just run the EXE program. It was tested with *Win98* and *XP*. Fig 5 shows a screen shot of the bridge; there is a lot of information on the screen. The monitoring scope with its controls is on the left side. The most important part is in the lower-right corner labeled UNKNOWN, where all of the relevant data about the measured impedance is displayed.

The bridge is easy to use but some considerations are in order. The value of *R*m must be known *exactly,* as all results depend on it. Strive for 1% accuracy here (or better) and do not use an inductive resistor; carbon or film types are fine. Although the bridge has a wide range, it is best to keep the levels of *V*x and *V*rm reasonable. The software scope helps monitor these levels. To reap the maximum benefit of the bridge, *R*m should be selected for the approximate range of impedance. For example, if *R*m is10 Ω, that is the approximate impedance to measure. This value can be chosen by trial and error or by making educated guesses, just as with most bridges.

In general, you should be able to measure over a 0.01 to 100 range based on *R*m. The chart in Table 1 illustrates the range you can expect for a given *R*m at 1225 Hz. (The bridge frequency will affect this range.) From the chart, if Rm =10 Ω, you can measure *L* between 12 µH and 129 mH, and C between 1299 µF and 0.129 µF at 1225 Hz. The software has provisions to store several values of Rm. Just make sure that is the value *actually* in the circuit.

At the high and low ends of the bridge's range, stray capacitance and inductance start to play a role. These values can be compensated out by us-

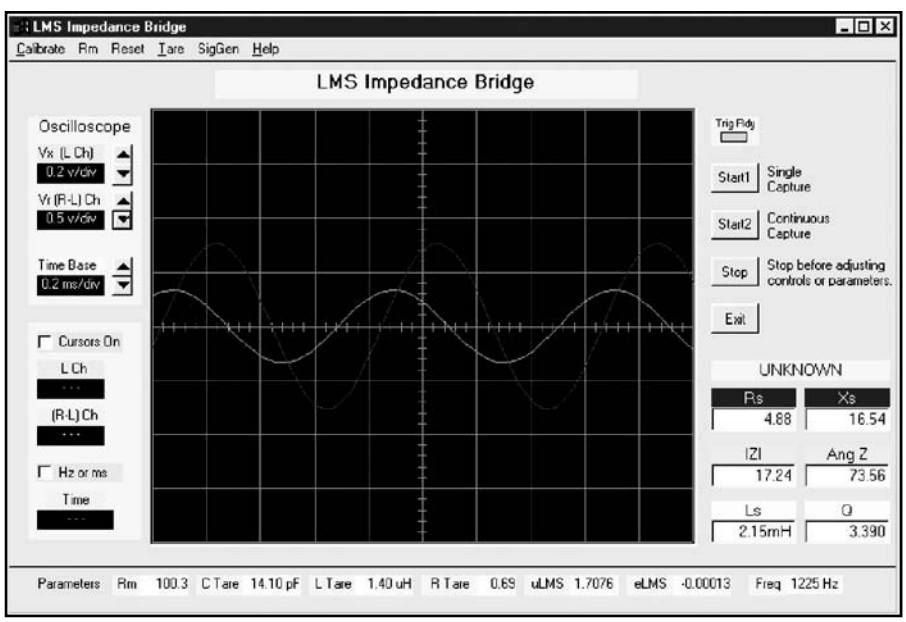


Fig 5—Main window of the LMS impedance bridge program.

Table 1—Range of Bridge for various *R*m with a Bridge Frequency of 1225 Hz. (See text)

*R*m (Ω)	L	C
10	12.99 µH to 12.99 mH	0.1299 µF to 1299 µF
100	129.9 µH to 129.9 mH	0.01299 µF to 129.9 µF
1 k	1.299 mH to 1299 mH	1.29 nF to 12.9 µF
10 k	12.99 mH to 12.99 H	1299 pF to 1.29 µF
100 k	129.9 mH to 129.9 H	12.99 pF to 0.129 µF

ing the bridge, itself, to measure them with the unknown impedance being an open circuit and short circuit, respectively. These values, called "tare," are then automatically used to compensate the result. For example, in my circuit there is about 14.1 pF of stray capacitance with *Z* open and *R*m = 100 kΩ. So, that value is what I enter for "C tare" in the program. Obviously, this only makes a difference when measuring small capacitors. Similar compensation may be made for the inductive wiring, when *Z* is near zero.

Once you have balanced the stereo channels and (optionally) calibrated the scope, you are ready to start measuring. By the way, all calibrations are saved so that you only need to do it once. Next, the mixer that came with your software or the one that came with *Windows* needs to be checked, as you may have changed its settings. When you start the program, a little notice comes on the screen to remind you about this.

Basically, you want to set the output level, input gain and stereo balance. The details of how to do this vary from system to system. Here is how it's done with the SB: In the mixer *Play* section, enable "Wave" and "Spkr," set the sliders to their maximums and mute all others (including "Line" to avoid audio feedback). In the *Record* section, enable "Line," set it to maximum and mute all others. Set the stereo balance to center for all controls.

Using The Impedance Bridge

Measuring impedances is very easy with this bridge, but it's helpful to think about what you are doing, so you don't misinterpret results. Don't try to measure a 100 pF capacitor with an *R*m of 10 Ω. Assuming that you have a "ballpark" *R*m, connect the unknown to the Z points of the circuit shown on Fig 2 and click the "Start" button on the screen. The bridge will automatically determine whether the reactance is capacitive or inductive at the measuring frequency. Several items are calculated and displayed, including the real and reactive parts of *Z*, the magnitude and angle of *Z*, the L or C value of the component and its Q or D factor. The scope is handy for looking at the relative magnitude and phase of *V*x and *V*rm and to see if you have reasonable levels.

If you aren't familiar with measuring impedance (and even if you are) you may run into some situations that are unusual or seem to give inconsistent values. As a guide, remember that the value of an impedance is usually defined only for a given frequency and may be different at other frequencies or signal levels. If you are measuring a resistor, you will find it has some reactance and it will show up in the box on the screen as either a capacitor or inductor. Since you know it's a resistor, just look at the magnitude of Z or real part of Z.

You are not limited to just L and C components; the bridge can measure all kinds of impedances including input impedance, audio transformer impedance, speaker impedance, solenoid impedance and even negative resistance. Some of these topics are covered in the following sections.

Measuring Input Impedance

The input impedance of an amplifier or other circuit may be found by connecting it to the unknown Z terminals as shown in Fig 6. Take care not to overdrive the amplifier input being measured. This can be controlled by adjusting the output level in the mixer and choosing a suitable value for *R*m. Note, too, that impedance may vary greatly with frequency, so try several bridge frequencies.

Here's an interesting special case: If you want to measure the input impedance of the left channel of your sound card line input, do the following. Bypass U1B and connect the lower end of *R*m directly to that channel input. This makes the input impedance the unknown *Z*. On my SB, the input impedance *Z*in measured 28.2 k ∠–6.41° at 1225 Hz (with *R*m = 1 kΩ).

Measuring Transformer or Speaker Impedance

The impedances of audio transformers or speakers can be found by simply connecting them to the bridge as shown in Fig 7. For speakers, use *R*m = 10 Ω to get started.

As an example, I measured the reflected impedance of a small audio transformer (Mouser 42MC003, 1.2 k: 8 Ω). I connected the primary to the unknown *Z* terminals with an 8.2 Ω resistor connected to the secondary, as shown in Fig 7. The bridge read 1.24 k ∠13.78 at 1225 Hz (with *R*m = 1 kΩ). The reflected impedance depends on the load connected to the secondary, so you can experiment with different loads to see the effect.

Measuring Large Electrolytic Capacitors

While I was working on this project, the power supply capacitor in my old oscilloscope went out. The replacement, a 1000 µF unit, was too great for my C meter. So, I put *R*m=10 in the bridge and measured it easily. Be careful with these kinds of capacitors and make sure they are discharged before measuring them.

Large capacitors can often have leakage and internal resistance. It's interesting to see how the capacitance of an electrolytic capacitor changes with frequency. A junk box capacitor measured 15.62 µF at 120 Hz, and it read 14.35 µF at 1225 Hz. By the way, it was marked 22 µF and thus outside its minus 20% tolerance.

Measuring Iron-Core Inductors

Iron-core inductors can give some strange results. First measure a small *air* core inductor (150 µH, or so) and vary the output level of the line by varying the volume out in the *Windows* mixer. Note that the reading for inductance barely changes; only due to noise and harmonic distortion errors. Now measure a small *iron*-core inductor and do the same thing. You will notice that the inductance decreases with signal level. This unexpected result (at least for me) required some research. It turns out that

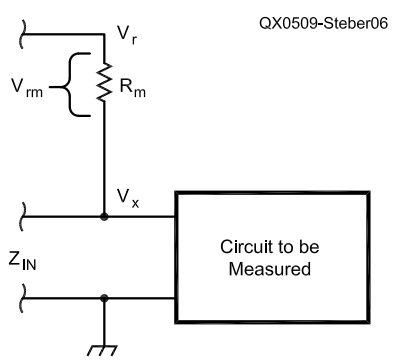


Fig 6—Setup for measuring circuit input impedance.

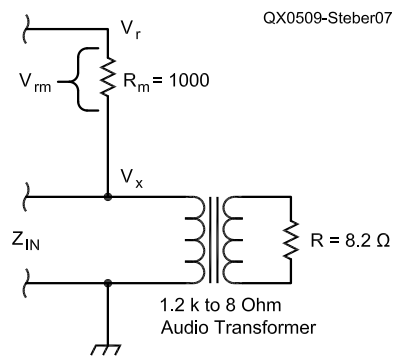


Fig 7— Setup for measuring transformer or speaker impedance.

the permeability of the iron starts out low for small currents and becomes higher and more constant (linear range) with increasing current. Since L is proportional to the permeability, there is a reduction in L at low currents. If you continue to increase the current *beyond* the linear range, the iron will saturate (I knew that!) and L will decrease. This last part is of concern to those who design switching regulators. Reaching such large currents is not within the capabilities of the bridge as it stands.

Another interesting thing about inductors is that they can have a resistive component that may vary with frequency. Using the transformer primary as shown above, with the secondary *open,* illustrates this. As the bridge frequency was varied from 525 Hz to 2205 Hz, the resistive part varied from 1.14 kΩ to 3.93 kΩ. Properly terminated with 8.2 Ω, it only varied from 1.06 kΩ to 1.26 kΩ over the same frequency range. Solenoids exhibit similar behavior and their impedance varies with plunger position.

Measuring Negative Impedance

It's possible to build a negative-resistance circuit. An article in *EDN Magazine* (Reference 5) about using negative resistance prompted me to do this experiment. Connect the negative impedance converter shown in Fig 8 as the unknown impedance *Z*. The op amp can be an LM358. Make sure R is less than Rm or else you will cancel it and cause oscillations. I used *R*m = 1000 Ω and *R* = 470 Ω. The bridge indicated *minus* 470 Ω. Be cautious with circuits of this type as they are likely to oscillate.

Final Thoughts

This program runs fine on *Win98* and *Win XP*. It was written in *Visual Basic* 6.0. When you run the software you may get a message like "Required DLL file MSVBVM60.DLL was not found." This is a *Visual Basic* run time file and is already on many systems. If it is not found, you will need to obtain it and install it on your system. It is freely available from Microsoft and other sites on the Web. It is usually available as *Visual Basic* 6.0 SP5: Run-Time Redistribution Pack (VBRun60sp5.exe) and is a self-extracting file. The download takes about six minutes at 28.8 kbps.

If you just want to experiment with the program, don't worry as it does not modify the registry or install any other material on your computer. You can remove it by just deleting the entire folder where it is located.

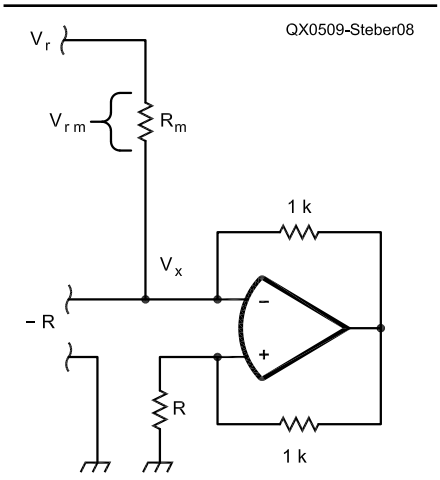


Fig 8—A negative resistance converter. Choose *R* < *R*m to avoid oscillations.

Numerous components were measured on a commercial LCR bridge and compared to the LMS bridge. Said LCR bridge was rated between 1% and 5% depending on range, component type and frequency. Good agreement was achieved between the two, better than 1% in many cases, a notable exception being small iron-core inductors (see comments above), which varied more. With any luck you will find similar results. In any case, don't be *impeded* in your quest for a fine LMS bridge. Perhaps this is one you can't *resist*.

George R. Steber PhD, is emeritus Professor of Electrical Engineering and Computer Science at the University of Wisconsin-Milwaukee. George, WB9LVI, is a life member of ARRL and was awarded the QST *cover plaque in May 1975 for a ground breaking article on digital slow scan TV. Dr Steber has considerable industrial experience as a corporate officer, consultant and product designer, with 18 patents issued. In his spare time he enjoys racquetball, reading, playing his Bach trumpet, editing video and astronomy. He recently restored a previously lost, badly damaged NBC* Tonight Show *program for the Kate Smith Commemorative Society. You may reach him at* **steber@execpc.com** *with "LMS" in subject line.*

Notes

[1]M. Dutta, et al, "An Application Of The LMS Adaptive Algorithm For A Digital AC Bridge," *IEEE Transactions on Instruments and Measurements*," Vol IM-36, pp 894-897, (Dec 1987).

[2]G. Steber, "Tracing Current and Voltage," *Circuit Cellar Magazine*, Vol #162, pp 56-61, Jan 2004.

[3]G. Steber, "Low Cost Automatic Impedance Bridge," *QST* (Oct 2005).

[4]B. Widrow et al, *Adaptive Signal Processing* (Prentice-Hall ISBN 0-13-004029-0, 1985).

[5]E. Simons, "Negative Resistance Load Canceller Helps Drive Heavy Loads," *Electronic Design Magazine*, March 19, 2001. ☐☐

Low Power DDS with the AD9834 and the Microchip PIC

The author presents the results of his examination of DDS circuitry for portable low-power operation.

By David Harrison, W6IBC

I've been experimenting with direct digital synthesis for several years. Analog Devices supplies a family of self-contained DDS chips and offers an excellent tutorial on DDS technology for download from its Web site (**www.analogdevices.com**). My first practical DDS efforts were motivated by a January 2002 *QST* article by Jim Hagerty, WA1FFL, describing a compact, direct digital VFO using a 90° quadrature encoder and an LCD display. WA1FFL offered a kit of parts including a surface mounted AD9835 DDS chip, controlled by a Phillips enhancement of the Intel 8051 microprocessor controller. I purchased several of the board kits and found that they worked quite well. Those boards required in excess of 100 mA of current and needed an external low pass filter. I didn't have access to, nor did I request a copy of, the WA1FFL microcontroller source code.

My next step in DDS technology led me to a series of Internet resources about an Analog Devices AD9850 DDS chip that is more capable than the AD9835. Beginning with an article by Curtis Preuss, WB2V, appearing in July/August 1997 *QEX*, a number of workers have programmed the Microchip PIC reduced instruction set (RISC) microcontroller (e.g. PIC16F84) to control the AD9850, and have placed their source code listings in the public domain for all to read and understand. I am most grateful for these public contributions.[1]

As part of my experiments with the AD9850, I taught myself how to program the Microchip mid-range PIC, essentially by following three paths. First, I acquired a PICKIT from Digi-Key, which contains excellent tutorial material as well as a PIC demo board and programmer. Second, I downloaded and studied the PIC-Elmer (PIC-EL) lessons provided over the last year by AMQRP, with many thanks to instructor John McDonough, WB8RCR. Third, I studied a highly useful text reference, "Programming and Customizing PICMicro Microcontrollers" by Myke Predko, McGraw-Hill. My efforts have enabled me to acquire sufficient fluency in assembly

312 Jessie Ct
PO Box 2298
Windsor, CA 95492-2298
w6ibc@arrl.net

[1]Notes appear on page 53.

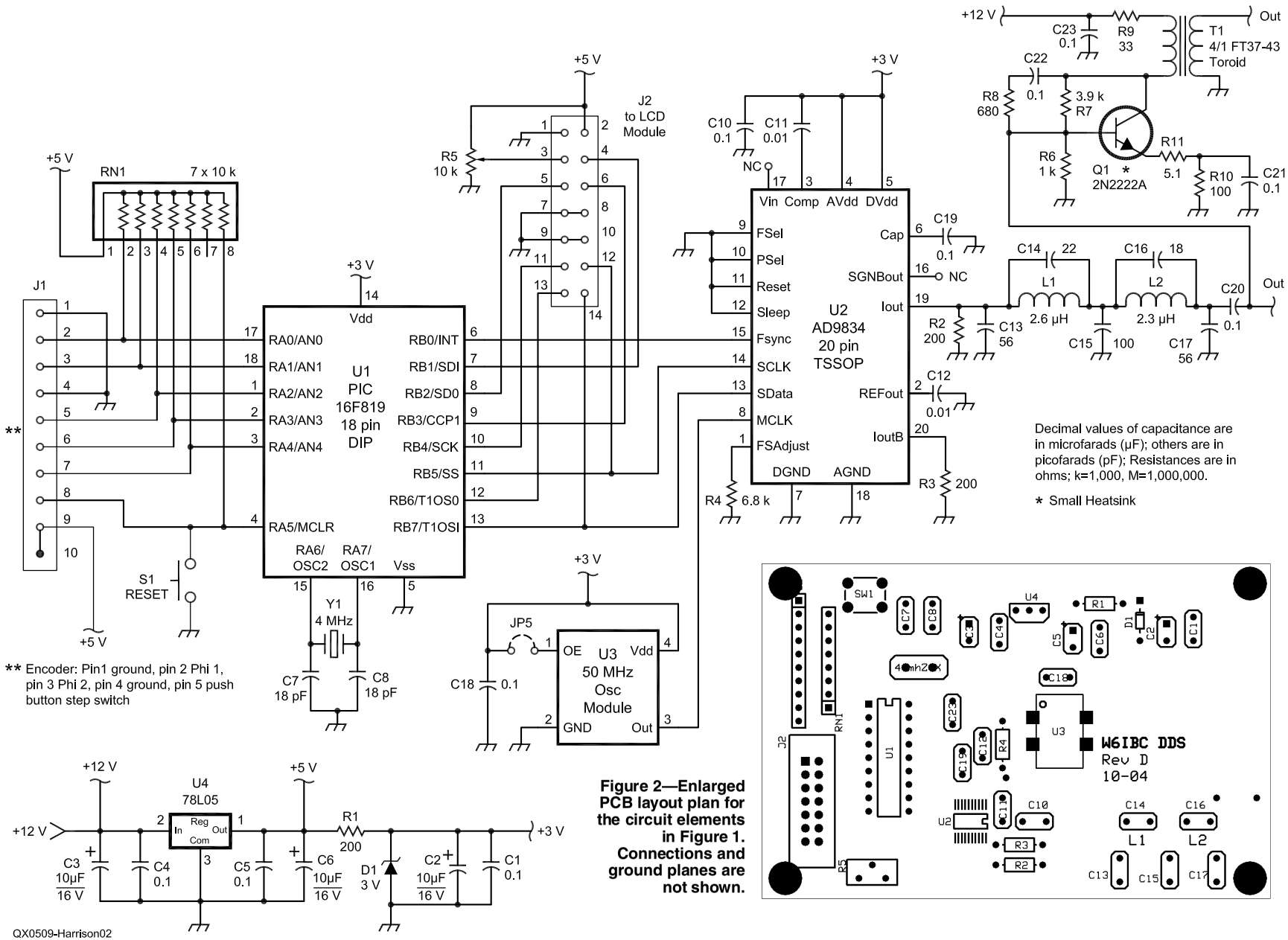


Figure 2—Enlarged PCB layout plan for the circuit elements in Figure 1. Connections and ground planes are not shown.

Figure 1—Circuit schematic and block diagram of a DDS board using the AD9834 (U2), clocked at 50 MHz by clock module (U3), under the control of a PIC16F819 (U1).

language programming to write and debug control programs for DDS using highly capable PIC midrange flash-reprogrammable microcontrollers including the 16F84, 16F628 and the 16F819. I also commend Microchip for its excellent MPLAB development environment, which is available by download from Microchip free of charge. I also constructed a PIC programmer driven from a serial port of my PC. A cautionary note, it took about one year of regular study and experiments with timers, LED flashers, keyers and other PIC projects for me to acquire a useful skill level for programming the PIC: having had no significant prior experience in assembly language programming (I had studied and used Dartmouth Basic in the 1970s as part of the personal computer revolution). This learning experience and effort has certainly been worthwhile from my perspective.

When Elecraft introduced its compelling KX1 QRP transceiver kit about a year ago, I downloaded the instruction manual and learned that the KX1 uses an Analog Devices low power DDS chip, the AD9834. After downloading the data sheet for the AD9834 from the Analog Devices Web site, I decided to attempt to migrate the preexistent AD9850 PIC control code to control the AD9834 with a mid-range PIC. In the end my efforts have proven successful. But, I encountered a number of pitfalls along the way, as I'll briefly describe. This project started with a detailed review of the AD9834 data sheet in comparison to the AD9850 data sheet. Of particular interest are the package dimensions, pinouts, and electrical and timing requirements for powering and controlling the AD9834.

All of the Analog Devices DDS chips are surface mount devices. The lead pitch of the AD9834 is particularly challenging at 0.65 mm. The AD9834 is packaged as a 20-pin thinshrink small-outline package (TSSOP). While there is at least one universal prototype board with 20-pin TSSOP pads that can be used to mount and connect the AD9834, in the end I decided I should make my own circuit board to accommodate the incredibly small leads of this part. Figure 1 is an electrical schematic of my board, and Figure 2 is a board layout, both crafted with software provided gratis by download from a board vendor, ExpressPCB. The board vendor's software enables one to draw a schematic, and also draw a board layout coupled to the schematic for verification. Once the board layout is completed and checked out, you can get a real time estimate of board cost from the vendor, and, if acceptable, upload your design and credit card info. If you do, printed circuit boards with plated-through vias made to your specifications arrive by mail in several days. The solution isn't particularly cheap, about $60.00 for three boards, but the alternatives I considered were less user-friendly. I'm sure other solutions are available.

Soldering the AD9834 to the minute printed traces takes patience. I tack one corner of the chip down to a few traces and then manipulate the device with tweezers and by using the soldering iron until all traces are aligned with respective pads. Then I flood both pad edges with solder. Using a high quality solder wick, I then remove excess solder from each row of pads and traces, and check with a magnifying glass and ohmmeter to be certain that no unintended solder bridges remain. Only after the installation passes the visual and electrical continuity tests, do I apply any power. I've used this method successfully with about a dozen parts by now. If you're careful it will work.

The board I designed includes the AD9834 (Digi-Key AD9834BRU-ND, $9.70), a low power 50 MHz crystal SMD module (Digi-Key 300-2207-1-ND, $2.81), an 18-pin DIP socket for the PIC16F819 controller (Jameco 223773CX, $3.59), an LCD header (surplus 16-character by 2-line LCD module from All Electronics Corp, about $9.00), a header for the shaft encoder (Digi-Key CT2999-ND, $5.40) and other buttons, a 5-V regulator, a two-stage elliptical low pass filter, and the necessary resistors and capacitors needed to complete the circuit. I also included a set of traces for a MMIC amplifier (e.g. Agilent INA-10386 from Dan's Small Parts) to increase the output level of the DDS chip to drive a passive diode ring mixer such as the Minicircuits SBL-1. However, the MMIC requires 45 mA of current for proper operation, not exactly low power. For a high level mixer such as the SA602, the MMIC was not used. My next board design (and present schematic) uses a 2N2222A wideband amplifier (to 30 MHz) drawing about 10 mA, rather than the 45 mA drawn by the 1GHz MMIC.

The AD9850 PIC controller code available via the AMQRP Web site links is very well commented, but it is written for the PIC16F84, an easy-to-understand and program chip with limited capabilities. The original program basically includes modules for initialization, reading the quadrature encoder (tuning knob), calculating a DDS control word and converting the DDS control word into binary coded decimal values for display on an LCD. In addition, it includes a calibration routine for calibrating the DDS oscillator against a standard such as WWV and EEPROM write/read routines for storing and retrieving the calibration values, and timing loops used primarily by the LCD routines. The original code included a routine for sensing the rate of rotation of the encoder and for increasing the tuning step size as rotational speed increases.

To have a low current solution, I chose a low power PIC, the 16F819. This chip has more memory and features than the 16F84, and requires a subtly different program sequence. A programmer specifically supporting the 16F819 will be needed to program this chip. I prefer *WinPIC*, freeware from DL4YHF (available at his Web site, **qsl.net/dl4yhf**). On the plus side, this chip will work with voltage levels as low as 2.5 V (as will the AD9834). I also chose a 50 MHz DDS SMD clock module that requires 3 V.

The software issues requiring resolution included determining the clock frequency word, initializing the AD9834, and converting a 32-bit DDS control word to the AD9834 28-bit format. Along the way, I added a positive IF offset capability with correction of the frequency display and implemented selectable fixed frequency step sizes. I selected nominal values of 1 kHz, 10 kHz, 10 Hz and 100 Hz, in that order, which I feel are better adapted to the detented electromechanical shaft encoder I selected for frequency control.

The AD9850 is basically a 32-bit device, meaning that its control word requires four bytes, plus a control byte. The AD9834 is a 28-bit device. It actually receives two 16-bit control words (four bytes), but the two highest order bit positions are control bits for selecting the destination registers, rather than high order bits of frequency-select data values.

The AD9850 data sheet defines the output frequency using the following formula:

$f_{OUT} = (\text{Delta Phase} \times \text{CLKIN})/2^{32}$

where:

Delta Phase is the value of the 32-bit tuning word,

CLKIN is the DDS input reference clock frequency in MHz, and

f_{OUT} is the frequency of the output signal in MHz.

The AD9834 data sheet gives the output frequency as above, except that

2^{32} is replaced by 2^{28}.

The first issue is how does one determine the four byte clock values for the AD9834? In the AD9850 code, the four-byte reference oscillator value represents the change in the frequency control word resulting from a one Hz change in output frequency. The reference oscillator value is interpreted as a fixed-point integer in the format (high byte) (next byte) (next byte) (low byte). To calculate the reference oscillator value the high byte is equal to 2^{32}/CLKIN. The next three bytes are the fractional part of (2^{32}/CLKIN) $\times$ 2^{24}. The results are expressed as eight hex values. For a 50 MHz clock the high byte is 0×55, and the next three bytes are $0 \times E6$, $0 \times 3B$ and 0×88.

The AD9834 uses a 28-bit control word rather than a 32-bit control word as used in AD9850, for example. Following the above algorithm in the same fixed point format, the high byte is equal to 2^{28}/CLKIN; and, the next three bytes are the fractional part of (2^{28}/CLKIN) times 2^{24} expressed in hex. For example, for a 50 MHz DDS clock module, the four byte reference oscillator value in hex for the AD9834 turns out to be 0×05, $0 \times 5E$, 0×63,and $0 \times B8$.

Once the oscillator reference value is established, the next issue is the serial control word format for controlling the AD9834. Three lines are needed to control both the AD9850 and the AD9834—a serial clock line, a serial data line, and a control line. But, there are important differences. The AD9850 expects the control line to be at a logical low. After 40 serial bits (5 bytes) have been received, the control line is brought high and then low, to latch the bytes into internal register locations. Thus, the output control routine for the AD9850 counts out 40 bits and then sends out a control bit.

The AD9834 receives serial data as 16-bit serial data-words from the microcontroller. The control line (Fsync) must be held at a logical high until a data-word is to be sent. Thus, during controller initialization, it is important to set the dedicated control line to a high level rather than merely clearing the port register. Otherwise, spurious data will reach the DDS. In addition, a 16-bit output routine must be provided in lieu of the 40 bit output routine used to control the AD9850.

Once the basic serial data format is worked out, the next problem lies with initializing the AD9834, which turns out to be far more involved than with the AD9850. Figure 8 of the AD9834 data sheet outlines the steps needed for software initialization of the chip. Once I worked my way through this maze, the simplest initialization routine I could dream up comprises the following sequence (written in PIC assembly code):

```
reset_DDS                       ; subroutine for resetting AD9834
        movlw       0×21        ; this is a control word that sets reset
        movwf       send1       ; and selects FReg0
        clrf        send0
        call        send_dds
        movf        AD9834_1,w  ; 14 LSBs of starting frequency in Reg 0
        movwf       send1       ; starting freq is 4.750 MHz
        movf        AD9834_0,w
        movwf       send0
        call        send_dds
        movf        AD9834_3,w  ; 14 MSBs of starting freq in Reg 0
        movwf       send1
        movf        AD9834_2,w
        movwf       send0
        call        send_dds
        movlw       0×80         clears FReg 1 LSB
        movwf       send1
        clrf        send0
        call        send_dds
        movlw       0×80        ; clears FReg 1 MSB
        movwf       send1
        clrf        send0
        call        send_dds
        movlw       0×C0        ; clears phase 0 reg
        movwf       send1
        clrf        send0
        call        send_dds
        movlw       0×E0        ; clears phase 1 reg
        movwf       send1
        clrf        send0
        call        send_dds
        movlw       0×20        ; Control word, clears reset
        movwf       send1       ; enabling DDS output of
        clrf        send0       ; starting frequency
        call        send_dds
        return
```

The above code snippet shows that the AD9834 must receive the low order 16-bit control/data word before it receives the high order 16-bit control/data word. The next issue relates to formatting these two control/data words. It turns out that one can continue to use the 32-bit multiplication routines for the AD9834 that were used to derive the frequency control words for the AD9850, with some necessary format alterations. The lowest order byte product of multiplication may be sent without being modified. The next order byte (MSB of low-order word) is modified. The highest two-bit positions are converted into a two-bit register select sequence. The low-order control/data word is then ready to be sent. Before this manipulation, the MSB of the low order word is saved so that the highest two bit positions of the calculation can be rotated into the lowest two bit positions of the LSB of the high order word.

After left rotation by two positions, the LSB of the high order word is ready to be sent. The MSB of the high order word is modified in the same manner as the MSB of the low order word. The following is my subroutine in PIC assembly language for converting the 32 bit multiplication product into the two 16-bit control words each having the required 14 bits of data and the two high order control bits:

```
; Handle low order byte of LSB word.
        movf    AD9834_0,w              ; this lowest order byte is unmodified,
        movwf   dds_0                   ; and can be used directly.
; Modify high order byte of LSB word by adding 2-bit preamble.
        movf    AD9834_1,w
        movwf   osc_temp_1              ; using osc_temp_1 for temp storage
; Add two control bits to high byte of LS word.
        bsf     osc_temp_1,6            ; force bit 6 high
        bcf     osc_temp_1,7            ; and bit 7 low, sel FReg0 of AD9834
        movf    osc_temp_1,w
        movwf   dds_1                   ; modified high byte of LS word now in dds_1
; Handle high order byte of MSB word by left shifting the three
; high order bytes in AD9834_1, AD9834_2 and AD9834_3 through
; the carry bit.
        movf    AD9834_1,w              ; get unmodified value
        movwf   osc_temp_1              ; store it back into osc_temp_1
        movf    AD9834_2,w              ; get the next unmodified value
        movwf   osc_temp_2              ; store in osc_temp_2
        movf    AD9834_3,w              ; get the high byte unmodified value
        movwf   osc_temp_3              ; store in osc_temp_3
; Now we're ready to rotate these three bytes two positions to the
; left to make up for 2-bit control values of dds_1.
        bcf     STATUS,B_C              ; clear carry flag of STATUS register
        rlf     osc_temp_1,f            ; rotate the three high order bytes two
        rlf     osc_temp_2,f            ; positions left, through carry flag of
        rlf     osc_temp_3,f            ; STATUS register.
        rlf     osc_temp_1,f
        rlf     osc_temp_2,f
        rlf     osc_temp_3,f
        bcf     STATUS,B_C
; Since we've moved the two high order bits from original AD9834_1
; to the two low bit positions of AD9834_2, we no longer need the
; osc_temp_1 value, so it can be ignored.
        movf    osc_temp_2,w
        movwf   dds_2
;       Now modify the high byte of the MS control word
        bsf     osc_temp_3,6            ; force high bit 6 to be 1 and
        bcf     osc_temp_3,7            ; high bit 7 to be zero, sel FReg0
        movf    osc_temp_3,w
        movwf   dds_3
        return
```

I also included new routines to limit the operating frequency to a range between 7.0 MHz and 7.3 MHz and a display subtract routine to handle a 12 MHz IF offset at the display. The calibrate routine was also modified to handle the initialization quirks of the AD9834. The EEPROM write and read routines were changed to meet requirements for the PIC16F819. Having made these changes and additions to the original AD9850 PIC control routines, it became possible to control and use the AD9834. I am certain that more experienced programmers working in the C programming language, for example, can provide more elegant solutions than the ones I am proposing. Yet, assembly language programming worked for me, and I guess that's one measure of success in a ham radio project.

Using the assembly code provided here,[2] I was able to realize a low power DDS controlled receiver for 40 meters, based on the Dave Benson receiver

board design described in Figure 17.86 of the 1995 ARRL "Radio Amateur's Handbook," in a breadboard lash-up shown in Figure 3 using the FAR Circuits receiver PCB. The PCB includes the receiver including an NE602 mixer, 12 MHz four-stage crystal filter, MC1350 IF amplifier, NE602 product detector, 12 MHz carrier injection oscillator, 2N7000 shunt muting transistor and LM386 audio amplifier chip. The local oscillator was not implemented and the DDS output was coupled directly to pin 6 of the NE602 mixer through a 0.01 µF ceramic dc blocking capacitor. Drive level from the DDS into the NE602 mixer was measured at about 100 mV p-p. The DDS board current (including LCD module) measured about 20 mA. An outboard FET preamplifier was also used in this setup, drawing an additional 8 mA.

While I prefer the much wider dynamic range available with a double balanced diode ring mixer, such as the SBL-1, the present setup without post-DDS amplification works very adequately to demonstrate the feasiblilty of DDS frequency control for small QRP radios, as clearly proven by the Elecraft KX-1. The PIC 16F819 is a very capable processor, with 2048 14-bit instruction words and control registers in four banks. It also has 256 EEPROM 8-bit register locations and up to 16 I/O pins (if the internal RC oscillator is enabled). There is plenty of room in the '819 for the DDS control and display process, as well as for other functions, such as a Morse keyer, transmitter offset and control, etc. I hope this will encourage others to experiment with DDS and microcontrollers, such at the PIC or the ATMEL Butterfly, etc. Improvements and adaptations will be obvious to the interested reader.

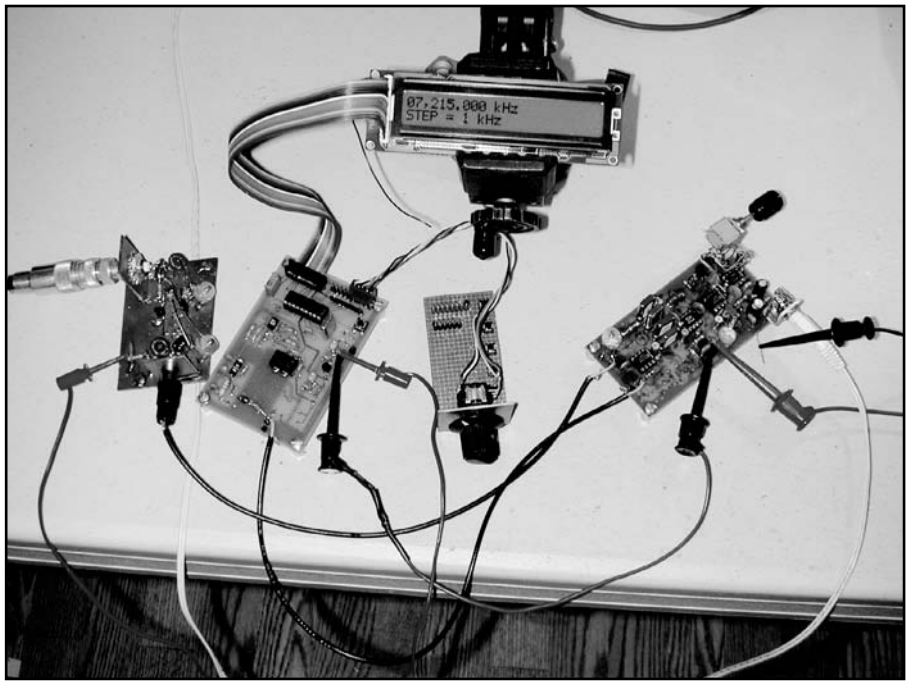


Figure 3—Breadboard lashup of modules including the boards of Figures 1 and 2. From left to right, a 40-meter FET preamp, the PIC-controlled DDS module, an LCD module mounted in a bench vise, a quadrature encoder mounted on a small multifunction carrier board, and the 4-chip 1995 ARRL Handbook 40 meter receiver module built on a FAR Circuits PCB.

Notes

[1]The interested reader should consult the American QRP club Web site (**amqrp.org**) that provides links to this work. In particular, I appreciated the NJQRP DDS daughtercard project and hope that this resource continues to be offered to the ham community. It provides an excellent first hands-on project with surface mounting and DDS technology. If one wants to experiment with the AD9850 and a PIC controller, I highly recommend using the DDS printed circuit board available from FAR Circuits, part *QEX* 7/97 Synthesizer $8.00 (**www.farcircuits.net**). Be sure to order the latest version of this board.

[2]The complete operating code is available on the *QEX* Web site, **www.arrl.org/qexfiles/9x05.harrison.zip**.

David Harrison, W6IBC, was first licensed as a teenager in 1957. He is a retired patent attorney who now has time to explore first hand some of the many technologies he wrote about on behalf of inventors during his career in Silicon Valley. He enjoys ragchewing with friends on 40 meter SSB, participating in local ARES activities, and building and testing ham radio equipment, particularly HF gear. He can be reached at **w6ibc@arrl.net**. □□

Antenna Options

By L. B. Cebik, W4RNL

Modeling Software, Part 1

A regular question that appears in my e-mail is what antenna-modeling program to buy or use. The question takes many forms, but the nub usually boils down to matching the options offered by one or more particular programs with the modeling needs of the potential user. So let's examine some of the alternatives in the available array of programs, beginning with the calculating cores that underlie the available implementations. In these notes, I shall assume that you are familiar with the basic terms of antenna modeling using round wires and can handle such terms as wire, segment, source, load, etc. If these terms are still a mystery, see the four-part series, "A Beginner's Guide to Modeling with *NEC*," *QST*, Nov 2000, through Feb 2001.

We shall discuss only *NEC* and *MININEC* modeling software, both of which are readily available in low cost or entry-level implementations. Table 1 provides a list of such software, along with Internet addresses for those who wish further information. The list is not absolutely complete, and it contains two entries that we shall not discuss. *Expert MININEC* is a proprietary revision of the original *MININEC* core, while *SuperNEC* uses a *MatLab* interface. Since I do not own either program, they will not figure significantly in the discussion. The "feel" of a program is a highly personal facet of modeling software. Therefore, examining the web sites of the program makers is the best route to determining if you will be comfortable using any of the programs. The web sites can also provide up-to-date information on features and price. Programmers are continuously adding features to programs, so these notes are technically out-of-date as soon as I write them.

1434 High Mesa Dr
Knoxville, TN 37938-4443
cebik@cebik.com

NEC* or *MININEC

There are two classes of cores that perform the "method of moments" calculations comprising the analysis of an antenna design. For a more complete history of the development of these two strains of cores, see Bob Haviland, W4MB, "Programs for Antenna Analysis by the Method of Moments," *The ARRL Antenna Compendium*, Vol 4, pp 69-73. *NEC* emerged from main-frame *FORTRAN* work, while *MININEC* was developed to work on early desktop computers having very limited memory resources. Both cores have undergone extensive upgrades. For example, reprogramming *MININEC* in one of the *Windows*-compatible languages has eliminated the early segment restriction on that core. Likewise, newer *FORTRAN* compilers for PC use have speeded up the runtimes of *NEC* models.

MININEC 3.13 is a public domain program. As a result, it has undergone significant modification to overcome some of its initial limitations in addition to removing the upper limit on the number of segments in a model. All of the *Windows* implementations listed in Table 1 have no upper limit on the number of segments that a model may have. Two older *DOS*-based programs, *AO* (or *MN*) by Brian Beezley, K6STI, and *ELNEC*, by Roy Lewallen, W7EL, did have limits of about 256 segments. *Expert MININEC* by EM Scientific employs a different algorithm set from the ones used in the public domain version.

The core of "raw" *MININEC* has a number of inherent limitations that may or may not be relevant to your modeling needs. All versions (with the exception of *Antenna Model*) use a ground simulation that becomes highly inaccurate if wires with any horizontal component are closer than about 0.2 λ above ground. All wires must be above ground, with only vertical wires allowed to touch the ground. The unmodified core is also sensitive to wire spacing, and such simple antennas as a folded dipole may give erroneous results. Sharp corners in the antenna geometry can also yield inaccurate results unless the wire segments are very short or the program has introduced a correction feature. Finally, the *MININEC* core has a frequency offset that becomes larger with increasing frequency. It becomes noticeable in the 10-meter region of the amateur spectrum.

Unlike *NEC* cores, which have undergone very limited modification, the *MININEC* core has seen extensive modification by individual programmers to overcome these limitations. Over a series of benchmark tests in which *NEC-4* has shown proven accuracy, the various implementations of *MININEC* show variable results, each according to the modifications introduced and

the success of those modifications. I ran all of the *MININEC* programs available to me through a series of benchmark tests, and only *Antenna Model* passed them all. (See "Testing the Fringes of Modeling Programs" at **www.cebik.com/amod/amod51.html** for details of the benchmark comparisons.) This fact does not mean that the other programs are not useful; instead, it means that they must be used within the frequency or structure limits built into them. There are many lower-frequency modeling tasks with relatively simple antennas for which any of the *MININEC* implementations will work well.

What *MININEC* does very well that *NEC* does far less well is handle wire junctions with different diameters, whether linear or angled. Indeed, the Leeson correction process, which some implementations of *NEC* include, used *MININEC* as the standard during its development.

NEC cores use a quite different set of algorithms. The cores will handle over 10,000 segments without modification, although software vendors may set a segment limit to various versions of the programs they sell. In addition, *NEC* has a number of additional features, the most prominent of which is the Sommerfeld-Norton (S-N) ground simulation routine. The S-N ground system is highly accurate, even for wires only fractionally above the surface. (Among *MININEC* implementations, only *Antenna Model* has grafted the S-N ground calculation system to its core.) The compiled *FORTRAN* routines used by many implementations are much faster for similarly sized models than most of the *MININEC* cores. In addition, *NEC* includes both networks and (lossless) transmission lines as part of the non-radiating accessories to models. In general, *NEC* cores have become a *de facto* standard for round-wire antenna modeling. *NEC* has a number of unique outputs (relative to *MININEC*), such as near-field analysis, received currents and scattering patterns from both linear and elliptical plane-wave excitation, and mutual coupling between specified wire segments. *NEC* also supports true ground-wave analysis that includes near fields and surface waves.

A full *NEC*-core implementation includes a considerable collection of geometry commands that permit the construction of complex antenna and allied structures with minimal input file size. There are commands to create arcs, circles, helices, and catenary wires. Another command permits both length and diameter tapering along a specified wire. There is a command for rotating, replicating, and moving a wire already created. A large rectangular wire-grid structure might take as few as four entry lines for a model in the format shown in Fig 4. *NEC* also includes provision for the creation of surface patches.

The key limitation of *NEC* cores is the inability to precisely handle elements having a changing diameter at wire junctions. For linear elements having no loads, transmission lines, or networks, the Leeson correction system has worked very well. (See David B. Leeson, W6QHS, *Physical Design of Yagi Antennas* (ARRL Antenna Book, 1992, Chapter 8, for details.) However,

Table 1—Some Available Programs.[1]

NEC-4	
GNEC	**www.nittany-scientific.com**
EZNEC Pro/4	**www.eznec.com**
NEC-2	
NEC-Win Pro	**www.nittany-scientific.com**
NEC-Win Plus	**www.nittany-scientific.com**
EZNEC	**www.eznec.com**
EZNEC Plus	**www.eznec.com**
EZNEC Pro	**www.eznec.com**
NEC2GO	**www.Nec2Go.com**
4NEC2	**www.qsl.net/wb6tpu/swindex.html**
SuperNEC	**www.supernec.com**
MININEC	
Expert MININEC	**www.emsci.com**
Antenna Model	**www.antennamodel.com**
NEC4WIN	**www.orionmicro.com**
MMANA	**www.qsl.net/mmhamsoft**
(This list does not include the *DOS* programs *AO/MN* and *ELNEC*.)	
Multi-NEC	**www.qsl.net/ac6la/index.html**

[1]For others, see **www.cebik.com/model/nec.html.**

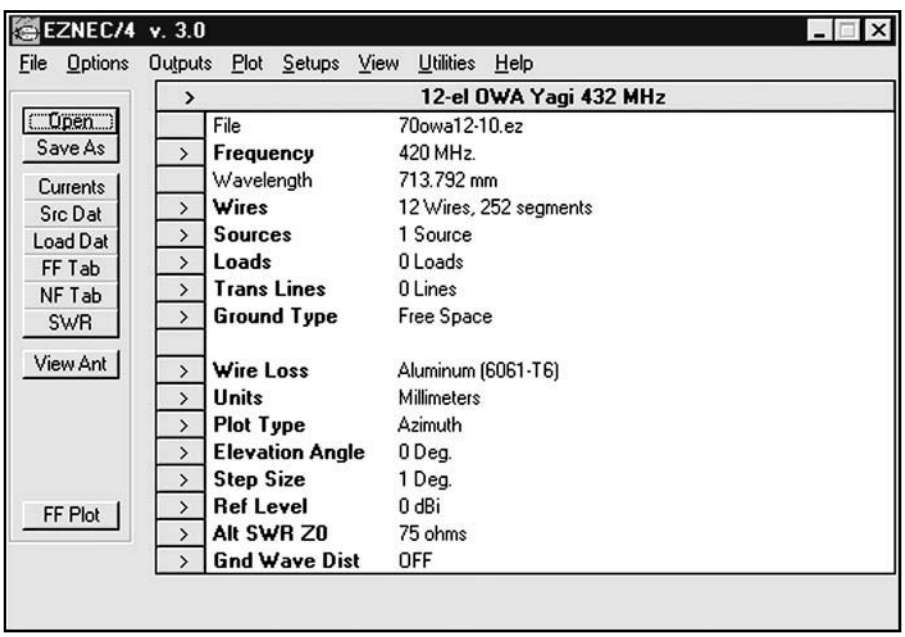


Fig 1—The *EZNEC* main screen.

if loads, lines, or networks are present along the element length or where the element is not linear, these correctives do not apply. Both *EZNEC* and *NEC-Win Plus* include the correctives.

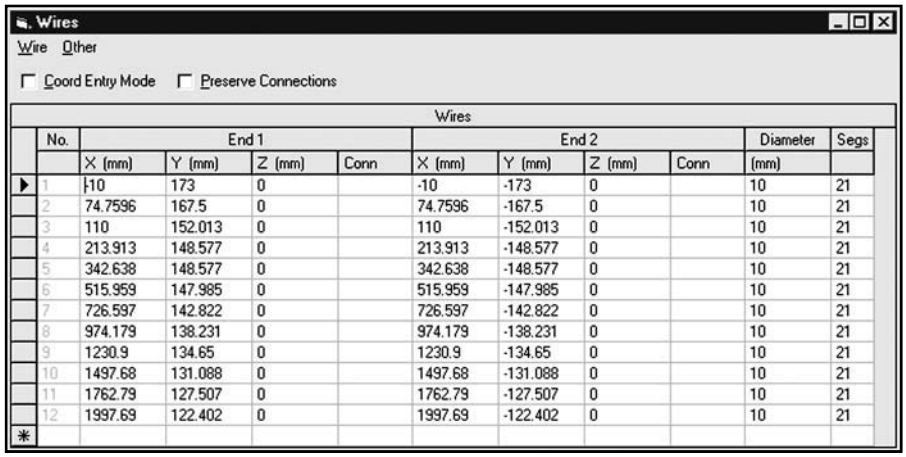
NEC comes in two generally available versions: *NEC-2* and *NEC-4*. (There is also a *NEC-3* that saw limited distribution.) We shall discuss the differences between these cores, but first, we should pause to look at file formats.

File Formats

One limitation that likely has limited the use of *MININEC* among antenna modelers is the fact that there is no standard file format. Fig 7, Fig 8, and Fig 9 present the main screens of *Antenna Model*, *NEC4WIN*, and *MMANA*. They suggest—correctly—that each program uses a different file format that is not directly convertible from one program to the next. In addition, the user must plug some model parameters directly into the program, although some implementations attach these parameters to the individual model file. As a result, it is generally the case that the modeler has to re-construct a model from scratch when moving from one program to the next.

Unmodified *NEC* cores require the use of an ASCII file of the sort shown on the main *GNEC* screen in Fig 4. (The *NEC-Win Pro* main screen would be virtually identical.) The model file contains all parameters of the model in terms of the geometry that describes the wires, the modifications to the geometry (such as the specification of element material loading and a source or excitation), and requests for outputs (the RP command in the illustration). The user may create a full model on a text editor as an input to the *NEC* core. The standard input file extension is .NEC, although any text editor, such as *Notepad*, will read the file and permit editing.

A number of *NEC* implementations use proprietary model file formats that are not ASCII. For example, the *NEC-Win Plus* files (see Fig 3) use a spreadsheet format. *EZNEC*, shown in Fig 1 and Fig 2, uses its own file format (and transfers data to its cores in binary form). Unlike many other programs, *EZNEC* uses separate screens for the main data and the individual collections of data that describe the wires, loads, sources, ground values, and other constituents of a model. *NEC2GO* uses a basic file system derived from, but not identical, to that


Wires

Wire Other

Coord Entry Mode Preserve Connections

Wires

No.	End 1				End 2				Diameter	Segs
	X (mm)	Y (mm)	Z (mm)	Conn	X (mm)	Y (mm)	Z (mm)	Conn	(mm)	
1	-10	173	0		-10	-173	0		10	21
2	74.7596	167.5	0		74.7596	-167.5	0		10	21
3	110	152.013	0		110	-152.013	0		10	21
4	213.913	148.577	0		213.913	-148.577	0		10	21
5	342.638	148.577	0		342.638	-148.577	0		10	21
6	515.959	147.985	0		515.959	-147.985	0		10	21
7	726.597	142.822	0		726.597	-142.822	0		10	21
8	974.179	138.231	0		974.179	-138.231	0		10	21
9	1230.9	134.65	0		1230.9	-134.65	0		10	21
10	1497.68	131.088	0		1497.68	-131.088	0		10	21
11	1762.79	127.507	0		1762.79	-127.507	0		10	21
12	1997.69	122.402	0		1997.69	-122.402	0		10	21

Fig 2—The separate *EZNEC* wire structure table.

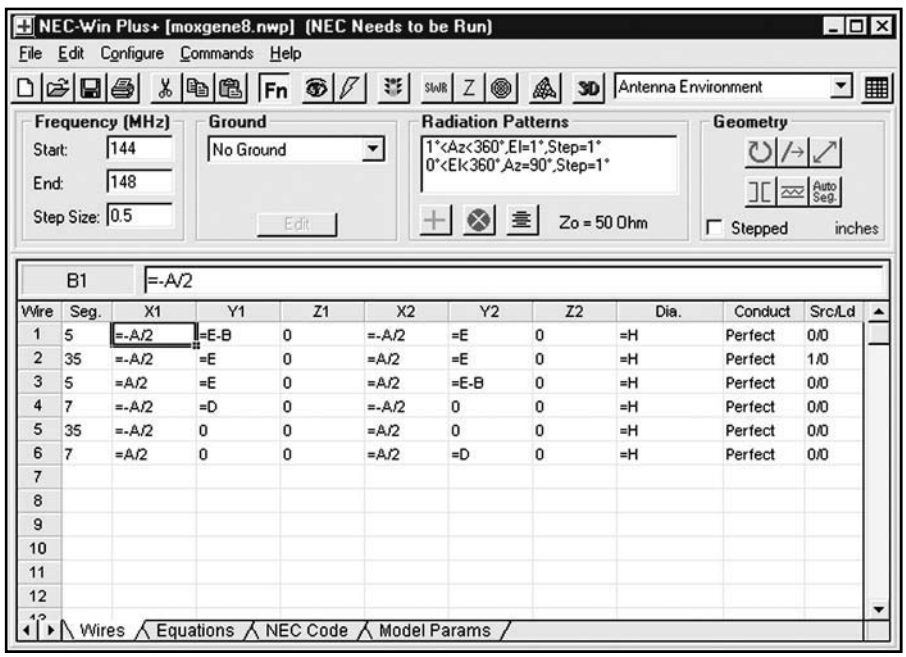


Fig 3—The *NEC-Win Plus* main screen.

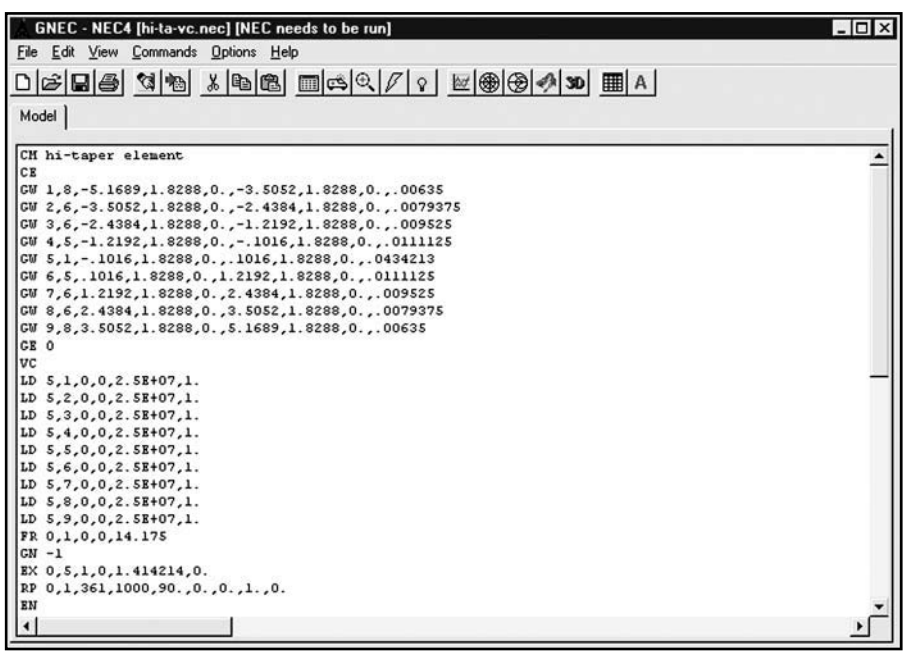


Fig 4—The *GNEC* main screen.

of *AO* and *NEC-Wires*. Nevertheless, all three programs—either in all or in advanced versions—include facilities for converting files constructed in their native file formats into a .NEC file. *4NEC2* (Fig 6), *NEC-Win Pro* , and *GNEC* save files in the basic .NEC format. The bottom line is that a file created under one system with a *NEC* core is convertible into a file to be saved under another system. *MultiNEC*, illustrated in Fig 10, while not a *NEC* program in itself, can read many formats, including those of *NEC-Win Plus*, *EZNEC*, and the *MININEC* program, *Antenna Model*. *Multi-NEC* is an *Excel* application that works with the core of an existing modeling program and adds a considerable number of features on both the input and output side of the core.

NEC-2 or *NEC-4*

The relative popularity of *NEC* as the basic calculating core for round-wire antenna modeling has resulted in the development of a larger body of modeling assistance than is available for *MININEC*. Perhaps the most notable training aid is the ARRL Antenna Modeling Correspondence Course, which comes with exercise models in *EZNEC*, *NEC-Win Plus*, and basic .NEC formats. However, other volumes (such as *Basic Antenna Modeling: A Hands-On Turotial*) and article series (for example, "Antenna Modeling" that appears monthly in *antenneX*) are available.

NEC-2 is the most widely used core and has become a public domain item. It is therefore available worldwide. The basic algorithms for the core treat only the axial currents, a fact that provides some of the core's limitations. It will not register the influence of a boom that intersects element wires at right angles. It requires a set of substitute uniform diameter elements for accurate calculation of linear elements having a changing diameter along their length. Non-radiating loads, transmission lines, and networks are most accurate in regions of an element that have a high and stable current, and such additions to the wire structure become less accurate as one moves into regions in which the current level changes more extremely from one segment to the next. As we bring wires of different lengths and diameters into close proximity, the accuracy of the result may suffer. All wires must be above ground.

Fortunately, *NEC*-2 contains a self-testing ability that will detect many model inadequacies. The average gain test is a necessary but not sufficient test of adequacy, and there are some inadequate models that the test will not detect. Although the convergence test is applicable to *NEC* models, it is the most used test for determining the adequacy of *MININEC* models. (*Antenna Model* incorporates the average gain test into its implementation of *MININEC*.) Hence, modelers who use *NEC* have a way to determine to a high, although incomplete, degree of confidence the quality of their models.

NEC-4 represents a further development of and revision to *NEC-2*. The current calculations use a different algorithm that gives higher accuracy to antenna structures that use a tapering diameter. However, for very steep tapers, the results do not fully mesh with *MININEC* or Leeson correction results. Since the new algorithms treat only axial currents, some

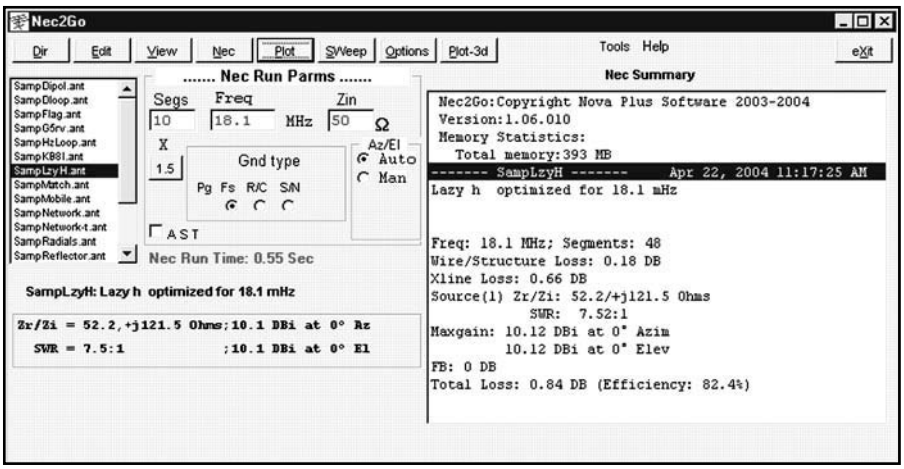


Fig 5—The *NEC2GO* main screen.

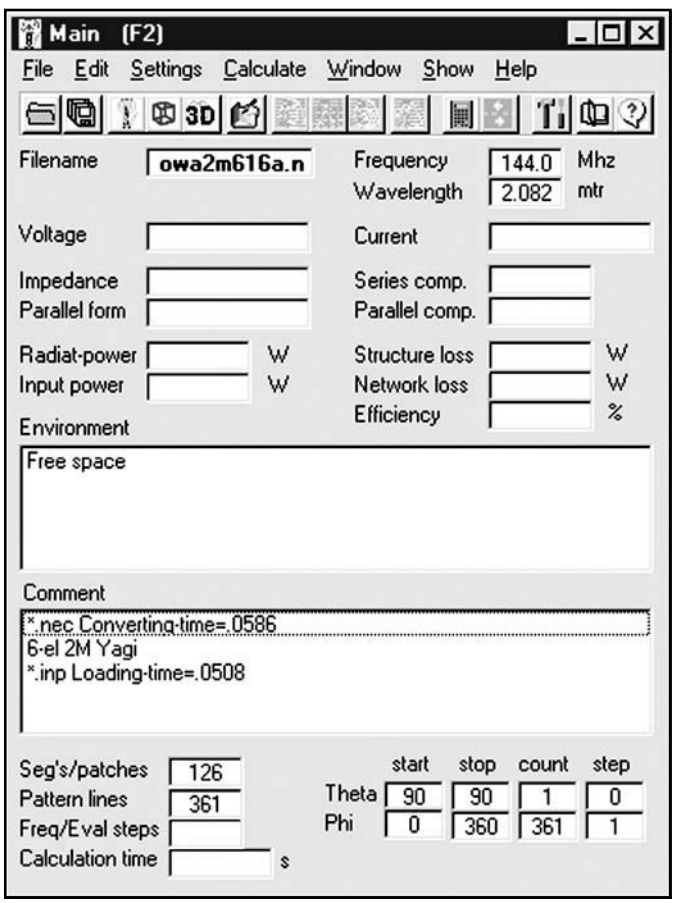


Fig 6—The *4NEC2* initial screen.

of the same limitations affecting *NEC-2* still apply, although in some cases, to a lesser degree. For further details, see "*NEC-4.1*: Limitations of Importance to Hams," *QEX*, May/June, 1998, pp 3-16.

NEC-4 adds a considerable number of new features that enhance modeling. The core permits wires below ground for accurate modeling of ground radial systems and similar subterranean structures. The core allows the modeler to specify insulating sheaths for wires. Besides the near-field analysis available in *NEC-2, NEC-4* also uses a second form of near-field analysis along the axis of a line specified by the modeler. Although the standard above ground medium is a vacuum or dry air, *NEC-4* allows specification of a different upper medium, with constant supplied by the modeler. The newer core also permits plot data file storage directly rather than via facilities provided by the commercial implementation.

NEC-4 has another important difference from *NEC-2*—it is not in the public domain. Instead, it is proprietary with the Lawrence Livermore National Laboratory and the University of California. It requires a separate license before a commercial implementation (*GNEC* or *EZNEC Pro/4*) can be sold. In recent years the cost of a license to an individual—such as a radio amateur—for non-commercial purposes has come way down, but may still be significant in deciding whether to invest in a *NEC-4* package. At the time of writing, the current cost of licensing is $250 for academic or non-commercial users and $950 for a commercial executable license. You may obtain the license materials on line at **www.llnl.gov/IPandC/technology/software/softwaretitles/NEC.php**. Export restrictions may apply, limiting access to the *NEC-4* core for users outside the United States.

For many applications, there are "work-arounds" available so that *NEC-2* results will reasonably replicate what you would obtain using *NEC-4*. However, for many other types of applications, *NEC-4* is necessary. For example, simulating a buried radial system for a vertical monopole was once believed to be possible using either a *MININEC*-type ground (available in *EZNEC*) or a system of radials placed very close to the ground. Subsequent modeling in *NEC-4* using buried wires has shown some serious shortcomings on the *NEC-2* work-arounds. Hence, for critical applications, *NEC-4* is the core of choice, if available.

Throughout these initial descriptions of both *NEC* and *MININEC*, I have referred to "round-wire" modeling. Both cores use algorithms based on the thin round wires, normally in a vacuum or dry air and well separated from other materials. A considerable amount of current antenna design uses elements with other cross-section geometries and other environments. For example, laying (or etching) copper strips on a substrate is a common construction method for antennas in the UHF region and upward. Without extensive external calculations of the adjustment for the changed geometry and the substrate, *NEC* cannot accurately model such structures. Those structures require the use of what some call hybrid programs that combine method-of-moments techniques with other means of accounting for altered current distribution and the influence of the base material. For even the well-heeled amateur interested in modeling, the cost of such programs—virtually all of which are proprietary—can be

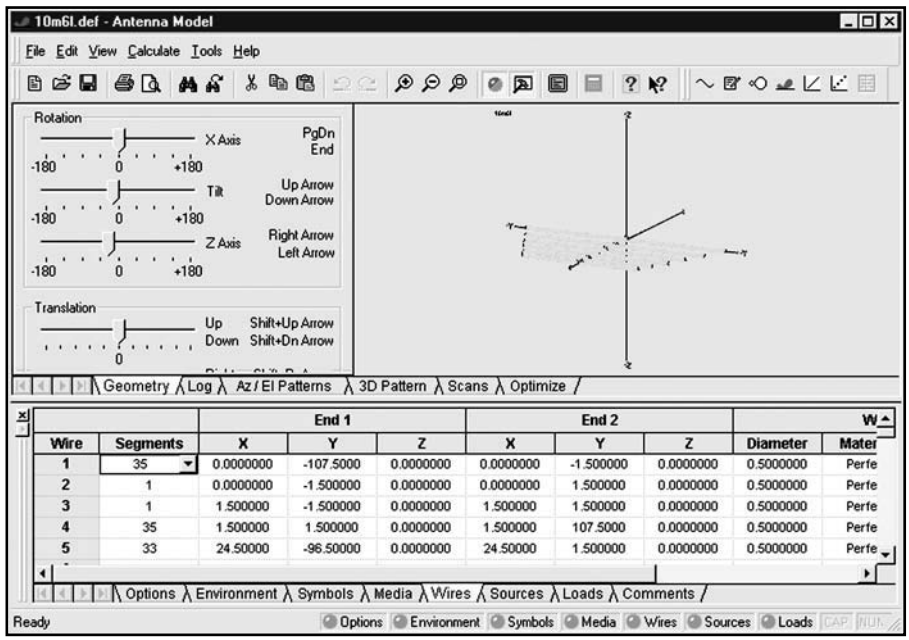


Fig 7—The *Antenna Model* main screen.

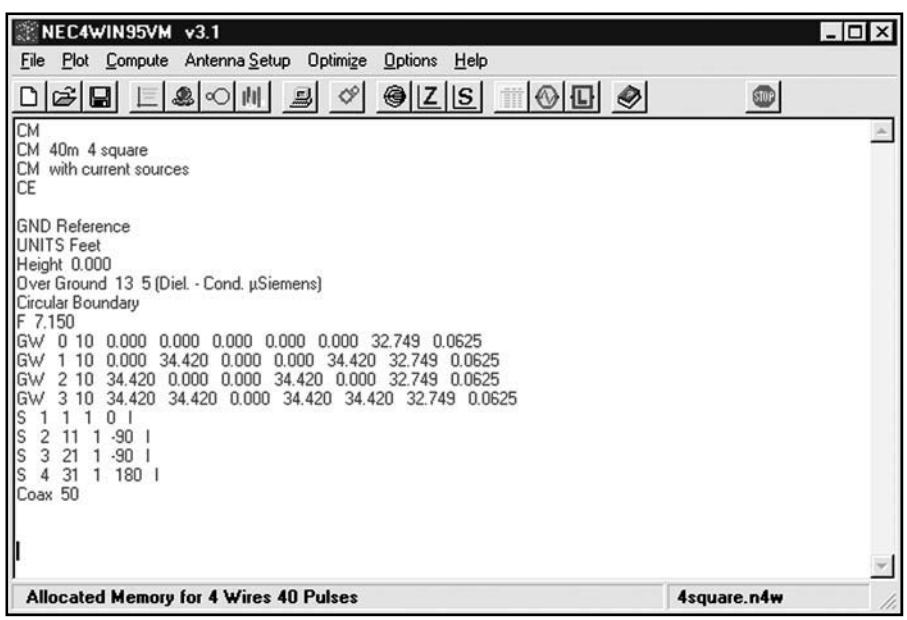


Fig 8—The *NEC4WIN* main screen.

daunting, if not downright forbidding.

Segment Limitations

Thus far, we have been looking primarily at differences among the calculation cores available for round-wire antenna modeling. The major implementation-specific differences that we have explored are the file formats for storing antenna models. However, among the commercial implementations of *NEC*, there is another more fundamental limitation—the number of segments permitted by the core. The procedure—from a programming per pective—for setting the maximum number of segments allowed is straightforward. Therefore, some commercial versions of *NEC* have segment limits below the maximum possible values.

The most notable of programs with segment limitations is *EZNEC*, now in version 4. Regular *EZNEC* limits the number of segments to 500, while *EZNEC Plus* allows 1500. *EZNEC Pro* (in both *NEC-2* and *NEC-4* versions), allows up to 20,000 segments by controlling the utilization of virtual memory. An allied decision made by commercial implementations of *NEC* is whether to use a single-precision or double precision core. Single precision cores run faster, although double-precision results are normally more precise. The speed difference does not show up in small models, while the precision difference does not appear until a model becomes highly complex.

NEC2GO claims its *NEC-2* core has no segment limitation. In contrast, the *NEC-2D* (meaning double precision) core in both *NEC-Win Plus* and *NEC-Win Pro* set the limit at 10,000 segments. *GNEC*'s *NEC-4D* core allows up to a little over 11,000 segments, since it does not internally control the use of virtual memory during a run. In addition, cores may automatically set their dimensions, including the allowable number of segments, memory and wires to a junction, by programming pre-sets, or by manual user settings. *EZNEC* sets the segmentation limits and memory allocation automatically by virtue of the model size, and *NEC-Win Plus* uses a similar system. However, *NEC-Win Pro* and *GNEC* have a parameter file to allow the user to set values for the most efficient core operation.

When new to antenna modeling, 500 segments may seem to be enough for the biggest imaginable problem. However, it is wise to consider both present and future uses of a modeling program before opting for the least expensive and most segment-limited version. Of course, economics may prove to be a decisive factor in decision-making, in which case, one may have to obtain simply the best program for the price. If you can take future potential modeling activities into account, the following brief examples may be of interest.

- Adhering to appropriate modeling guidelines for either *NEC* or *MININEC*, a 5-band quad may require as many as 220 segments per set of loops for each band.
- A 5-λ, 3-wire rhombic may require up to 600 segments.
- A 160-meter 4-square array with a full buried radial system with 32 radials per monopole may require nearly 1600 segments.
- A VHF or UHF corner reflector of considerable size and composed of a wire-grid structure may require over 2200 segments.

How large your models might someday become as you refine your modeling efforts presents you with some interesting match-ups with available software.

I began these notes with the vain hope of compressing the options facing the potential antenna modeler into a single session. However, that goal is not realistic, so we shall have to spend one more episode on the subject. With the transition into options involving the number of available segments, we are moving from general considerations of the *NEC* and *MININEC* cores into choices among the features and facilities offered by the available implementations. We shall continue in this vein next time. □□

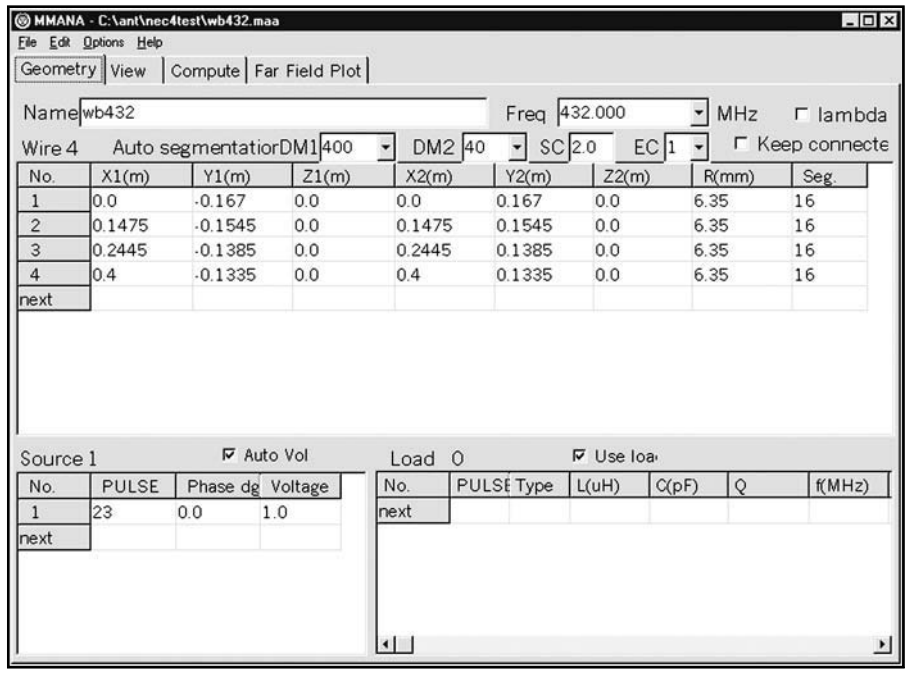


Fig 9—The *MMANA* main screen.

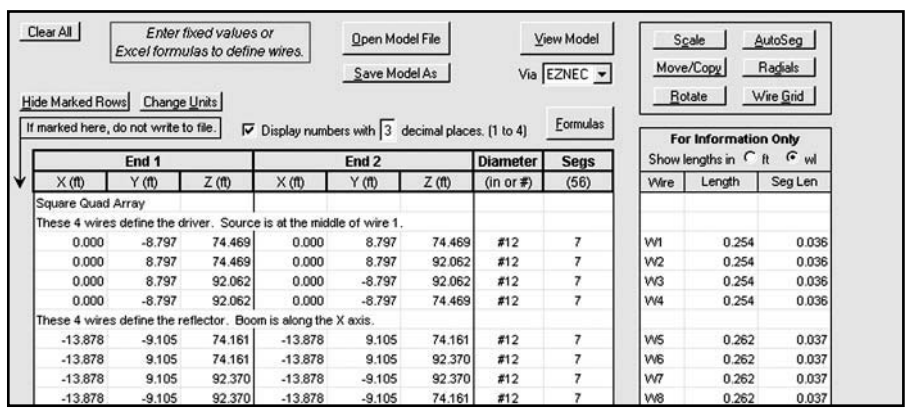


Fig 10—The *Multi-NEC* initial screen.

Letters to the Editor

ATX Adventures (Nov/Dec 2004)

Dear Mr. Smith,

Phil Eide, KF6ZZ, wrote a wonderful article on converting an ATX power supply into a high-current 13.8 V supply for amateur use. His reverse engineering of the original ATX supply design was amazing, and it revealed a robust design. I was somewhat dismayed at his conclusion from his failure analysis. What with over-stressed, underrated and just plain missing RFI chokes, no wonder the supply had died and destroyed his computer in the process! It is hard to avoid his conclusion that this was "a low-cost build of a fundamentally good electrical design." However, what disappointed me was that his failure analysis stopped short of an ultimate cause.

In order to sell electronic devices in the U.S. and Europe, a manufacturer must be in compliance with standards of quality (ISO-9000) and RFI/EMI (Part 15 and VDE). Many countries and regions of the world do not subscribe to these high standards, so a compliant design may be subjected to a "low-cost build" that produces a device which is supposed to work reliably only as long as it is in warranty. However, It is illegal to import such a device into the U.S., and since Mr. Eide's ATX supply is obviously such a device, it must have been imported illegally, and thus is a counterfeit. This is not the first time I have heard complaints about such power supplies. Just a few months ago, a letter in *QST* ("Taming computer power supply noise," "Hints & Kinks," *QST*, July 2004, p 65) described retrofitting a similar supply with the RFI chokes that had been left out. There is enough RF pollution in this country without equipping PCs with RFI generators.

So, what is needed here is reporting and enforcement by FCC and Customs. I would like to see ARRL organize a program that trains hams to look for counterfeit power supplies and provides them with contact information to report these supplies.

Respectfully submitted—*Avery Davis, WB4RTP, 969 N Miller Dr, Tucson, AZ 85710;* **wb4rtp@mindspring.com**

"A Blind Automatic Frequency Control Algorithm" (Jul/Aug 2005)

Hi Doug,

My compliments to Gary, WAØSPM, on his recent article on "blind" AFC control for SSB, in Jul/Aug 2005, *QEX*. I do have a question about it, however. In the article, Gary states that the noise of the IF amplifiers and detectors needs to be sampled at 16 kHz, minimum, to minimize aliasing. My question is:

Since the highest usable information frequency is 3 kHz in this scheme, why not limit the baseband audio to slightly over 3 kHz with an elliptic sharp cutoff analog filter and sample at 6 kHz or if using a sound card, 8 kHz? This would satisfy the Nyquist criteria, eliminate aliasing, and improve the overall noise performance of the system. Am I missing something? *73—Pete McNulty, WA1SOV, 8 Settlers Ln, Sandy Hook, CT 06482-1312;* **wa1sov@earthlink.net**

Pete,

Using a calculation very similar to yours, I originally wrote the code to run at an 8 kHz sample rate. When I started testing the algorithm, several issues came up that forced the change to a higher sample rate.

The first issue revolved around the combination of the filters in the receivers and the sound card CODEC. In some test cases I detected aliased products in the FFT results. This is similar to the "monkey chatter" problem that has been reported in some DSP receivers. To completely eliminate this, it is necessary to attenuate signals at ½ the sample rate and above, to have an amplitude of less than ½ of the least significant bit of the A/D converter used. In this case (8 kHz sampling with a linear 16 bit A/D), that's about 100 dB of attenuation at 4 kHz and above. I just couldn't guarantee this with some receivers I tried. With the 22,050 Hz sample rate, the combination of the IF filter in the receiver and the input filter to the soundcard did the trick in all of my tests; I easily had more than 100 dB of attenuation at 11,025 Hz and above.

I also observed that the performance improved as I increased the number of vocal harmonics included in the computation. While this wasn't a driver for SSB in the ham bands, many of my tests were run against commercial AM stations because they have good frequency stability and repeatability.

If you plan to run this algorithm on high-quality SSB gear that contains good skirts and filters with good rejection above 4 kHz, you may be able to get away with the 8 kHz sample rate. As you mentioned, there are real advantages to keeping the sample rate as low as practical. Thanks for the comments!—*73, Gary Geissinger, WAØSPM; 680 N Gooseberry Ct, Lafayette, CO 80026-1524;* **ggeissinger@digitalglobe.com**

Octave for Signal Analysis (Jul/Aug 2005)

Doug:

There are a couple of important pieces of information missing from this article. Aside from those, it is a very interesting and useful article.

I suspect I am, like most readers, and not familiar with *Octave*. My first reaction (after reading the code) to Maynard's suggestion of using ifft() to reproduce the incoming signal was that it most certainly would not give back the original (in the general case) if all that was present in the frequency domain vector was magnitude information.

I downloaded *Octave* 2.1.50 for *Windows,* ran some simulations and read the documentation carefully.

The first missing piece is that the abs() function is not the same function that it is in *FORTRAN* or *C*. This function operates only on complex numbers and produces the magnitude of a complex number rather than the absolute value of a scalar. It turns out that scalars are a special case of a complex number and the function produces the same results. The name is most certainly confusing if you are not aware of the intent of the language author. This explains why his frequency-domain plots show the power spectrum (as on a spectrum analyzer) rather than the Fourier Transform.

The second piece of missing information is that the fft() function produces a vector of complex numbers. If you plot s_cxr without using the abs() function you will see very small values for the real part of the Fast Fourier Transform for all of his examples. All of the energy (except for some leakage due to the errors in a FFT) is contained in the imaginary part of the vector and would not be plotted by: plot(time1, s_cxr); —*Regards, Ray Mack, WD5IFS,* QEX *Conributing Editor,* **rmack@arrl.org**

On Digital Video

Hi Doug,

You may want to look into the work of F4DAY, who is doing interesting experiments with amateur digital video broadcasting. There are a few competing approaches, but most of them are just a pre-built board with a lot of preprogrammed chips. Jean-Francois, in contrast, tries to explain and come up with something useful using "kitchen-table" technology. **perso.wanadoo.fr/jf.fourcadier/** or **perso.wanadoo.fr/jf.fourcadier/index_e.htm**.

I'm not sure if you've heard, but some time ago Jan-Martin Noding, LA8AK, passed away. His achievements in Amateur Radio may warrant mention in *QEX*. I would be surprised if he never wrote for *QEX* but can't check. In any case, details at **www.agder.net/la8ak/**.—*73, Geert Jan de Groot, PE1HZG;* **pe1hzg@arrl.net** □□

In the next issue of QEX/Communications Quarterly

In the next *QEX*, Rod Brink, KQ6F, describes his new direct-conversion phasing rig for 75 m and 40 m. The unit uses a Stamp II module for control and an AD9835 DDS for tuning. Rod combines modern subsystems to achieve an up-to-date version of an old idea. □□

Upcoming Conferences

MICROWAVE UPDATE 2005, October 27-October 30, 2005

Presented by the San Bernardino Microwave Society and the Western States Weak Signal Society at the Sheraton Cerritos Hotel Towne Center, 12725 Center Court Dr, Cerritos, California; tel (562) 809-1500 • fax (562) 403-2080. Visit **www.microwaveupdate.org/** for more information.

2005 AMSAT-North America Space Symposium and Annual Meeting, October 7-9, 2005, Lafayette, Louisiana

Lagniappe: (pronounced "lon-yop") comes from the rich Creole dialect of southern Louisiana, or Acadiana, and represents the generous Cajun tradition of always giving "an extra or unexpected gift or benefit." This year's AMSAT Space Symposium is sure to deliver. The 2005 Symposium will be held in Lafayette, Louisiana, on the weekend of October 7-9.

Activities will include an authentic Creole shrimp boil and swamp tour, in addition to our annual banquet featuring local Creole cuisine and an exciting guest speaker.

The city of Lafayette is the hub of Acadiana. It is a focal point for the region, with fine dining, music, arts, advanced medical facilities, numerous retail shopping centers, the Cajundome and the University of Louisiana at Lafayette. Lafayette is conveniently situated 35 miles north of the Gulf of Mexico, 50 miles west of Baton Rouge, 129 miles west of New Orleans and only 229 miles east of Houston.

Registration

For the first time, attendees can now register online through the AMSAT store at **www.amsat-na.com/SymposiumReg.php**.

Hotel Information

The 2005 Space Symposium will be held at the Holiday Inn Central, Lafayette's only Holidome. The Symposium room rate is $75 per night. The hotel is located in downtown Lafayette, at the intersection of I-10 and I-49. Holiday Inn Central offers free high-speed Internet access, an indoor pool and sauna, whirlpool, electronic game room, exercise room, restaurant and lounge. It has a full-service restaurant and lounge open for breakfast, lunch and dinner. For additional information, visit their Web site at **www.amsat.org/amsat-new/AboutAmsat/symposium.php**.

Airport Information

Lafayette Regional Airport (LFT) is serviced by Delta, Continental and Northwest Airlines, with connections to LFT are through Houston, Dallas/Fort Worth, Memphis and Atlanta. For more information, visit the Lafayette Regional Airport_Web site (**www.lftairport.com/traveler_information.html**). More direct flights are available to Baton Rouge, approximately 60 miles from Lafayette, and New Orleans, approximately 130 miles away.

Train and Bus

Lafayette is serviced by AMTRAK and is a stop on the Sunset Limited (Los Angeles to Orlando.) Trains run east and west approximately every 3 days. Lafayette is also serviced nationally by Greyhound Bus Lines. The Lafayette Transit System (LTS) runs local bus service from 6:30 AM to 11:30 PM. Fares generally run $0.75 for adults and there are child and senior discounts. A day pass is available for $2 per day.

State and Local Tourism

Lafayette offers much for out-of-town visitors, from music that is sure to get your blood pumping to cuisine that is a connoisseur's delight. There is much about Acadiana just waiting to be discovered.

2005 ARRL/TAPR Digital Communications Conference, September 23-25, 2005

The 24th Annual ARRL and TAPR Digital Communications Conference, will be held in Santa Ana, California. The ARRL and TAPR Digital Communications Conference is an international forum for radio amateurs to meet, publish their work, and present new ideas and techniques. Presenters and attendees will have the opportunity to exchange ideas and learn about recent hardware and software advances, theories, experimental results and practical applications. Topics include, but are not limited to:

- Software defined radio (SDR)
- Digital voice
- Digital satellite communications
- Global Position System (GPS)
- Precision timing
- Automatic Position Reporting System (APRS)
- Short messaging (a mode of APRS)
- Digital Signal Processing (DSP)
- HF digital modes
- Internet interoperability with Amateur Radio networks
- Spread-spectrum
- IEEE 802.11 and other Part 15 license-exempt systems adaptable for Amateur Radio
- Using TCP/IP networking over Amateur Radio
- Mesh and peer-to-peer wireless networking
- Emergency and Homeland Defense backup digital communications
- Using *Linux* in Amateur Radio, updates on AX.25 and
- Other wireless networking protocols.

Visit **www.tapr.org** for more information.

Mid-Atlantic States VHF Conference, September 24, 2005

This is sponsored by the Mt Airy VHF Radio Club, Saturday, Sep 24, 2005, at the Courtyard Marriott (3327 Street Rd, Bensalem, Pennsylvania; tel 215 639-9100; **www.bensalempacourtyard.com**/. There is a special Conference rate of $99/night that includes a breakfast coupon. An advance registration ($69) includes: conference registration, *Proceedings*, Friday evening hospitality, a Saturday morning coffee break, Saturday lunch, auctions, Saturday evening banquet and a ticket to the HAMARAMA radio hamfest on Sunday, Sep 25, nearby at the Wrightstown Grange.

Visit the Web site for further information and maps: **members.ij.net/packrats/hamarama**. Visit Philadelphia's historic attractions: the new Liberty Bell and Constitution Center. Atlantic City and Cape May, New Jersey and the Poconos are also great visitor destinations. □□

Get the card that helps keep you on the air

Introducing the ARRL Travel Rewards Visa® Platinum Card!

Show your ham radio pride with the new ARRL Travel Rewards Visa Platinum Card! You receive great Platinum benefits each time you use it. And every purchase supports ARRL programs and services—at no cost to you!

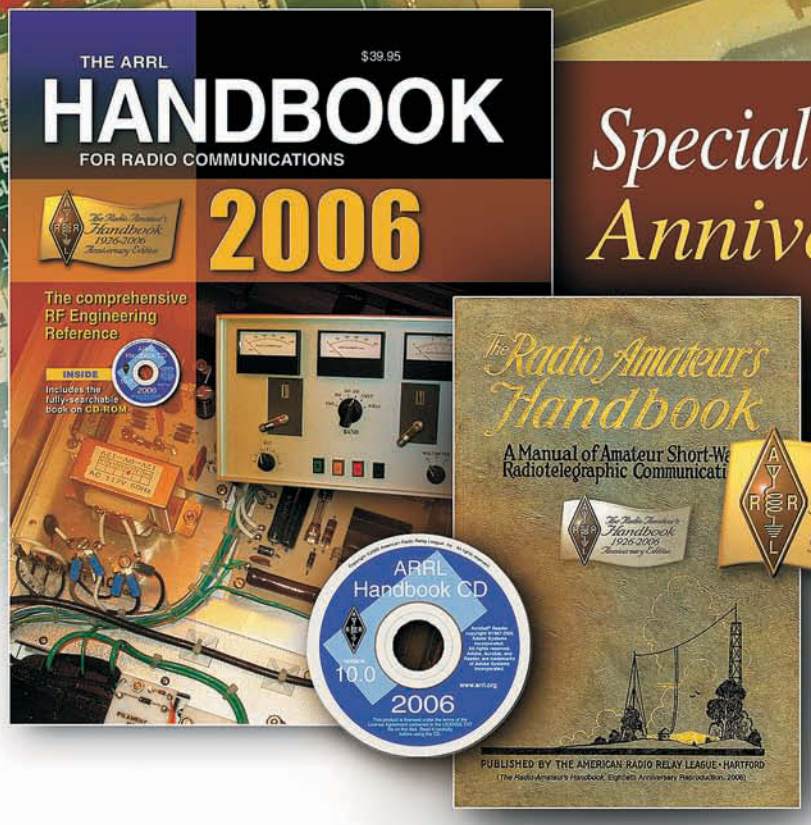
- No annual fee
- Low introductory rate*
- No balance transfer fees for six months*
- Earn reward points with every net purchase
- Redeem your reward points for travel, merchandise and more!
- Each purchase benefits ARRL!

Demonstrate your commitment to Amateur Radio.

Apply today!

Complete an application,
Call **1-800-853-5576 ext. 8543** or
Visit **www.arrl.org/visa**

*Some limitations may apply.
U.S. Bank National Association ND is creditor and issuer of the American Radio Relay League Travel Rewards Visa Platinum Card.



Special ARRL Handbook Anniversary Offer!

Available from ARRL and Select ARRL Publication Dealers

Facts about The First *Handbook!*

- First edition, 224 pages, published by ARRL in 1926.
- Originally titled, ***The Radio Amateur's Handbook***.
- Authored by the late **Francis Edward Handy**, **W1BDI**, ARRL Communications Manager. The reproduction is a facsimile of the author's signed, personal copy, with minor handwritten notes.

Pre-order **The ARRL Handbook—2006 Edition** with Bonus "**1926 *Handbook***" Reproduction.

You get all this when you pre-order:

- **The ARRL Handbook—2006 edition.** Up-to-date and always revised!
- **The ARRL Handbook on CD-ROM—version 10.0.** included with every book.
- **BONUS "1926 *Handbook*" Reproduction.** Limited Edition, commemorating 80 years of **The ARRL Handbook**.

80 Years of Excellence!

Since 1926, generations of radio amateurs and experimenters have come to know **The ARRL Handbook** as THE standard in applied electronics and communications.

About the 2006 Edition

ALWAYS REVISED! This 2006 edition builds on the extensive rewrite of the previous edition, and includes the most up-to-date theory, references and practical projects (including weekend-builds!) for receivers and transmitters, transceivers, power supplies, RF amplifiers, station accessories, and antennas.

NEW HF AMP! An impressive addition to **The ARRL Handbook** is a *brand-new*, high-power HF linear amplifier project using the new Eimac 3CX1500D7 power triode.

CD-ROM INCLUDED! This edition is bundled with **The ARRL Handbook CD (version 10.0)**—the complete and fully searchable book on CD-ROM, including many color images, additional software and reference material.

Pre-order Today www.arrl.org/shop
or Toll-Free 1-888-277-5289 (US)

Softcover. Includes book, CD-ROM and Bonus "**1926 *Handbook***" reproduction
ARRL Order No. 9485..... **$39.95** plus s&h

Hardcover. Includes book, CD-ROM and Bonus "**1926 *Handbook***" reproduction
ARRL Order No. 9493..... **$54.95** plus s&h

Pre-order now and get the special Bonus "**1926 *Handbook***" reproduction! Offer expires September 30, 2005, or while supplies last. Pre-orders will ship after October 1.

Bonus "**1926 *Handbook***" reproduction offer expires 11:59 UTC, September 30, 2005 or while supplies last. A limited supply will be printed before it is returned to the "ARRL vault" for at least 10 years. Bonus "**1926 *Handbook***" premium cannot be redeemed for cash. No returns. Exchanges must be accompanied by premium. Valid only on pre-orders direct from ARRL or from Select ARRL Publications Dealers. This offer may be canceled or modified at any time due to system error, fraud or other unforeseen problem. Void where prohibited.

Shipping and Handling charges apply. Sales Tax is required for orders shipped to CA, CT, VA, and Canada.

Prices and product availability are subject to change without notice.

ARRL *The national association for AMATEUR RADIO*
225 Main Street, Newington, CT 06111-1494 USA

SHOP DIRECT *or call for a dealer near you.*
ONLINE WWW.ARRL.ORG/SHOP
ORDER TOLL-FREE 888/277-5289 (US)

QEX 9/2005