$5

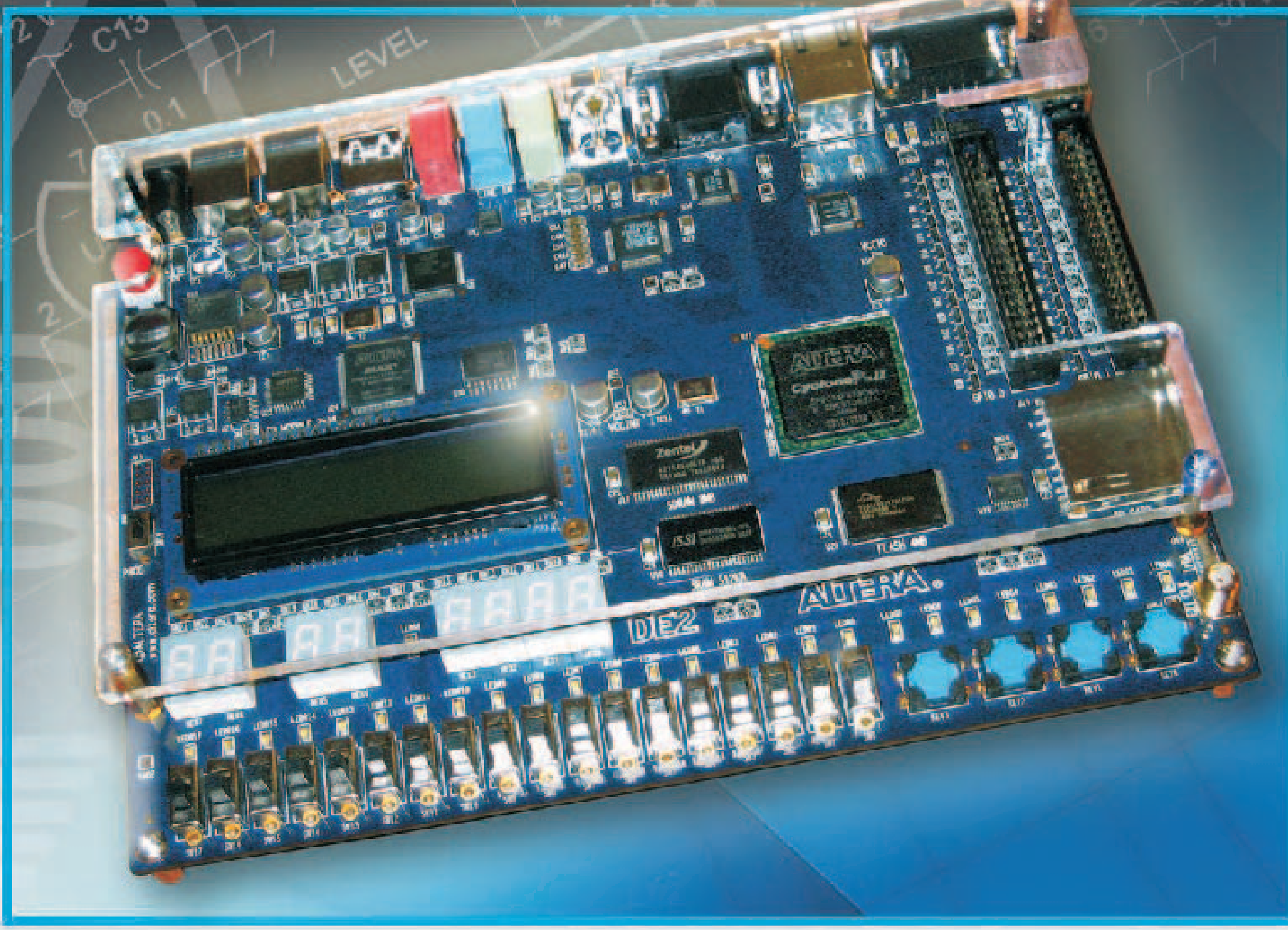
January/February 2010

www.arrl.org

# QEX

## A Forum for Communications Experimenters

Issue No. 258

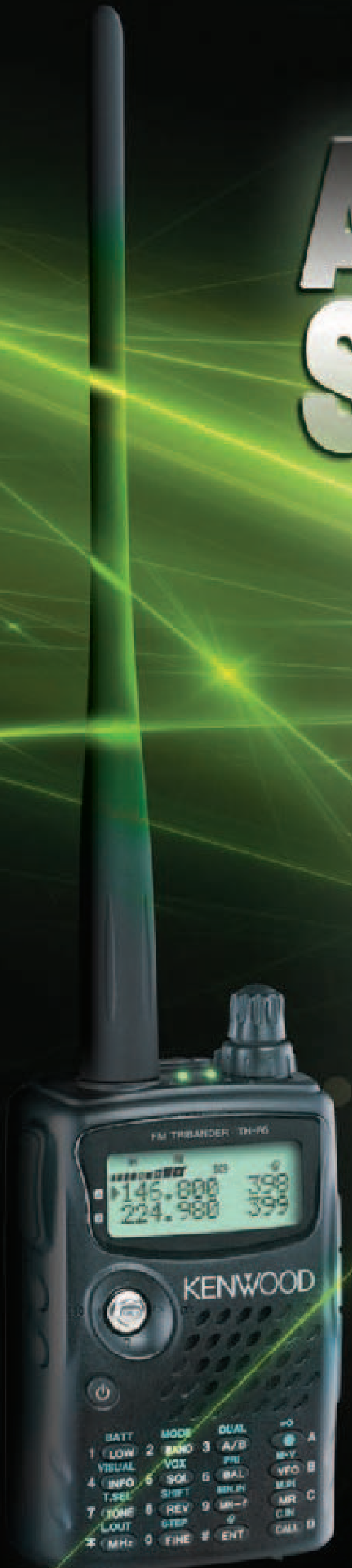



**KB1JIJ** describes a spread spectrum transmitter he developed as a US Military Academy digital logic class project. The transmitter is built on an Altera DE2 Development and Education Board with a Cyclone II FPGA.

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

KENWOOD
Listen to the Future

# AIRWAVE SUPERIORITY




**Never before has a compact HT offered as many features, and such high powered performance as the TH-F6A. Arm yourself with one today and gain your own airwave superiority.**

- Triband 144/220/440 MHz all at 5W
- Receives 2 frequencies simultaneously even on the same band
- 0.1~1300MHz high-frequency range RX (B band)[1]
- FM/FM-W/FM-N/AM plus SSB/CW receive
- Bar antenna for receiving AM broadcasts
- Special weather channel RX mode
- 435 memory channels, multiple scan functions
- 7.4V 2000mAh lithium-ion battery (std.) for high output[2] and extended operation
- 16-key pad plus multi-scroll key for easy operation
- Built-in charging circuitry for battery recharge while the unit operates from a DC supply
- Tough construction: meets MIL-STD 810 C/D/E standards for resistance to vibration, shock, humidity and light rain
- Large frequency display for single-band use
- Automatic simplex checker
- Wireless remote control function
- Battery indicator • Internal VOX • MCP software

[1]Note that certain frequencies are unavailable. [2]5W output

## TH-F6A
**TRIBANDER**

www.kenwoodusa.com

JQA-1205 091-A
ISO9001 Registered
Communications Equipment Division
Kenwood Corporation
ISO9001 certification

**KENWOOD U.S.A. CORPORATION**
**Communications Sector Headquarters**
3970 Johns Creek Court, Suite 100, Suwanee, GA 30024
**Customer Support/Distribution**
P.O. Box 22745, 2201 East Dominguez St., Long Beach, CA 90801-5745
Customer Support: (310) 639-4200 Fax: (310) 537-8235

ADS#36709

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 258

Harold Kramer, WJ1B
*Publisher*

Larry Wolfgang, WR1B
*Editor*

Lori Weinberg, KB1EIB
*Assistant Editor*

Zack Lau, W1VT
Ray Mack, W5IFS
*Contributing Editors*

**Production Department**

Steve Ford, WB8IMY
*Publications Manager*

Michelle Bloom, WB1ENT
*Production Supervisor*

Sue Fagan, KB1OKW
*Graphic Design Supervisor*

David Pingree, N1NAS
*Senior Technical Illustrator*

**Advertising Information Contact:**

Janet L. Rocco, W1JLR
*Business Services*
860-594-0203 – Direct
800-243-7768 – ARRL
860-594-4285 – Fax

**Circulation Department**

Cathy Stepina, *QEX Circulation*

**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;

International and Canada by Airmail: ARRL member $31, nonmember $43;

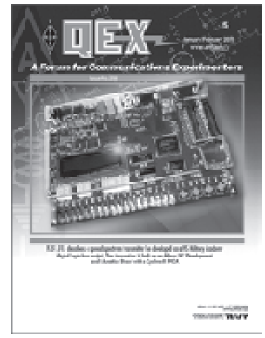
Members are asked to include their membership control number or a label from their QST when applying.

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




US Military Academy Cadet Thomas Dean, KB1JIJ, describes a spread spectrum transmitter he developed as a digital logic class project. Built on an Altera DE2 Development and Education Board with a Cyclone II FPGA, the transmitter demonstrates some principles of spread spectrum systems and software defined radio transmitters.

## In This Issue

# Features



# Columns



## Index of Advertisers

**The American Radio Relay League**

The American Radio Relay League, Inc, is a noncommercial association of radio amateurs, organized for the promotion of interest in Amateur Radio communication and experimenta-tion, for the establishment of networks to provide communications in the event of disasters or other emergencies, for the advancement of the 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 three 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 corres-pondence should be addressed to the administrative headquarters:

ARRL
225 Main Street
Newington, CT 06111 USA
Telephone: 860-594-0200
FAX: 860-594-0259 (24-hour direct line)

**Officers**

***President:*** JOEL HARRISON, W5ZN
528 Miller Rd, Judsonia, AR 72081

***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 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* or high-resolution digital images (300 dots per inch or higher at the printed size). 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.

**Larry Wolfgang, WR1B**

lwolfgang@arrl.org

# Empirical Outlook

**Where do we go from here?**

It's a simple question, really, and one I've heard many times over the years. You probably have, too. As we reflect on the past and look to the future, it's not unusual to wonder what that future holds. It would be nice to have a crystal ball and be able to see a year or two or ten years into the future. Of course for all the magic that is ham radio, that's not part of our bag of tricks.

In years past I have been in a number of meetings, discussing the future of Amateur Radio. I remember when packet radio was the hot new topic, and there was a lot of discussion about building packet networks to link all Amateur Radio operators through one or another networking plan. While the days of packet bulletin boards and that initial burst of activity may be waning, others have quietly gone on to build new network opportunities such as the APRS network, with its messaging opportunities and emergency communications capabilities. Many operators use WinLink and other message forwarding techniques on a daily basis.

Software defined radio has been a hot topic for many years now. We continue to see new developments and improvements in radio technology and capabilities. Most modern transceivers make use of at least some aspects of the SDR concept. While some of the best performing radios still rely on traditional radio circuitry, they also allow new software versions to be downloaded directly to the radio, with vast new capabilities and subtle performance enhancements. Other high performance radios have moved toward the concept of direct digitization of the RF signals, with all processing done in dedicated digital signal processing ICs and personal computers. Whether you want to buy or build a high-performance radio, there are plenty of opportunities to play with SDR.

Another buzzword I've heard a number of times is "adaptive radio." As I understand it, the concept is that the transceivers might be able to change their operating parameters automatically, based on band conditions and communications modes. Perhaps the radio detects interference from another signal near its operating frequency. Filter bandwidth and passband position shifts to cut out the offending signal. The radios communicate about received signal strength and other band conditions under the operators' communications and adjust transmitter power and perhaps even digital data speeds. All of this happens without any operator intervention. Some of this may sound a little farfetched, but is it really? What can we do to build more intelligence into our radios? What new features would you like to find on your next dream radio?

What other changes do you see on the horizon? Will we see years of subtle changes and improvements to our current circuitry and techniques, or is there some new idea ready to break through and change the way we define "radio" in the future? Are there radical new antenna designs waiting to be discovered, or do we know all there is to learn about antenna design? What new digital (or analog) communications techniques and modes will be invented or evolved? What do you think?

*QEX* has published the technical details of all these past developments, and continues to help advance the state of the radio art. Whether you want to learn about the latest techniques or want to experiment and push the envelope yourself, we want to bring you the information you need. I am sure we will continue to have articles about packet networking systems and software defined radios. There is also still plenty for most of us to learn about more traditional circuitry. We haven't seen the last of articles about crystal oscillators or direct digital synthesizers. We are sure to see more articles about a different approach to (or maybe a rediscovery of time-honored methods of) antenna design and construction.

I believe many of us want to read about your ideas of what the future will look like. What new circuit ideas are you experimenting with? What visions do you have for new radio/communications circuit methods and modes? Whether it is a pie-in-the-sky wish list of performance features or information about a circuit you have built, we would like to hear from you. After all, if *QEX* is going to bring new ideas to the world of Amateur Radio, we depend on you, our readers, to write the articles to present those ideas.

**Jon Wallace**

111 Birden St., Torrington, CT 06790, fjwallace@snet.net

# Amateur Radio Astronomy Projects

*The author participated in a variety of activities during the International Year of Astronomy in 2009.*

As my IYA 2009 (International Year of Astronomy) activities come to a close, I would like to share some of my favorite radio astronomy projects with you in the hope that you will enjoy them as much as I have.

I am a science teacher in Connecticut. I have long thought that too much stress is placed on visual science, and I've always tried to expose students to non-visual experiences. With this in mind, I started exploring radio astronomy in the early '80s and joined the Society of Amateur Radio Astronomers (SARA). I got lots of help building projects and have been at it ever since (check out SARA at **radio-astronomy.org**).

The first project I will discuss, a solar radio in the VLF range, is one I started many years ago. It is very simple and yet yields data that can be reported to a national organization and is used to do "real science!" It studies the reaction of Earth's ionosphere to solar activity by measuring the intensity of a received signal. These radios are known as SID (Sudden Ionospheric Disturbance) monitors, since they are measuring the changes in intensity of a radio signal due to ionospheric disturbance caused by solar flare activity.

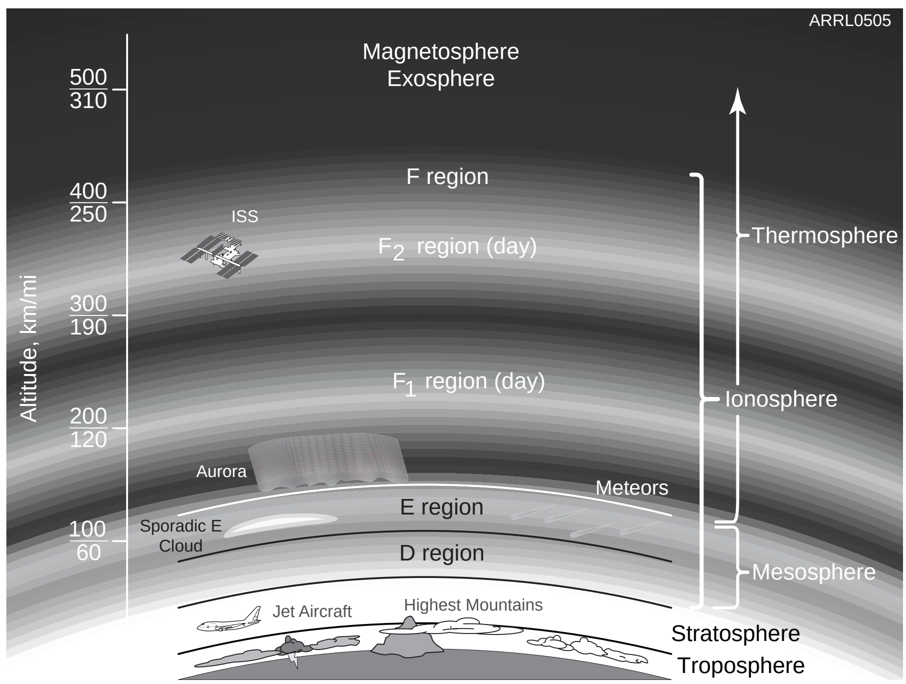



**Figure 1 —Earth's atmoshphere, including regions of the ionosphere.**

## Ionosphere

This region of the atmosphere is ionized by solar and cosmic radiation. It ranges from 70 to 1000 km (about 40 to 600 miles) above Earth's surface, and is generally considered to be made up of three regions D, E, and F. Some also include a C region and most experts split the F region into two (F1 and F2) during the daylight hours. Ionization is strongest in the upper F region and weakest in the lower D region, which basically exists only during daylight hours. This is because the number of free electrons increases as you rise through the atmosphere, reaching radio significance at about 70 km. You also have lower pressure at higher elevation. These conditions lead to the production of more monatomic gases and ionization that lasts longer because of the distance between atoms/molecules. Most of the ionization we are discussing is due to ultraviolet light, but at lower altitudes X-rays are needed to produce the ionization seen. Figure 1 is an illustration of the Earth's atmosphere, including the ionosphere.

## Solar Flares

Flares are enormous explosions that occur near sunspots on the surface of the sun, lasting roughly an hour and heating the region to millions of kelvins. See Figure 2. Most astronomers believe these events are caused by the sun's magnetic field. Since the sun is a fluid, magnetic field lines can get twisted. Near sunspots, the field lines get so twisted and sheared that they cross and recombine releasing an explosive burst of energy that can travel tens of thousands of miles off the surface of the sun. Obviously, as this energy makes its way to Earth, ionization in the atmosphere is greatly enhanced.

Flares are thus closely tied to the 11 year sunspot cycle. The sunspot cycle relates to a solar magnetic cycle, which runs for 22 years. During the first 11 years, sunspot frequency increases to a maximum and then decreases. At this point the magnetic poles flip polarity and the cycle begins again for the next 11 years. We are currently coming out of the Cycle 23 minimum and some sunspots from Cycle 24 have been detected. Figure 3 shows how the positions of sunspots on the Sun's surface create a butterfly pattern, with 12 sunspot cycles represented.

Flares that can be detected with VLF radios are generally caused by X-ray flares and have various flux levels associated with them. Figure 4 explains the classification of these flares.

The flares we can detect with VLF radios are C, M and X. C flares and below are fairly weak disturbances, with little effect on communications. M flares are medium sized flares that can cause short periods of radio blackout and minor radiation storms. X flares are large events that cause major planetary blackouts and radiation storms. When I'm not sure whether or not I've detected a flare, I always check "The Solar Events Report" at the NOAA Space Weather Prediction Center (**www.swpc.noaa.gov/ftp-menu/indices/events.html**). Here they list events by day and time, which allows you to check your results.

### VLF and the Ionosphere

During the daylight hours, VLF signals generally pass through the D region and are refracted by (or reflect off) the E region, thus leading to a weakened signal. During a flare event, the D region is strengthened and acts as a wave guide for VLF signals, since the wavelength of the signal we are monitoring is a significant part of

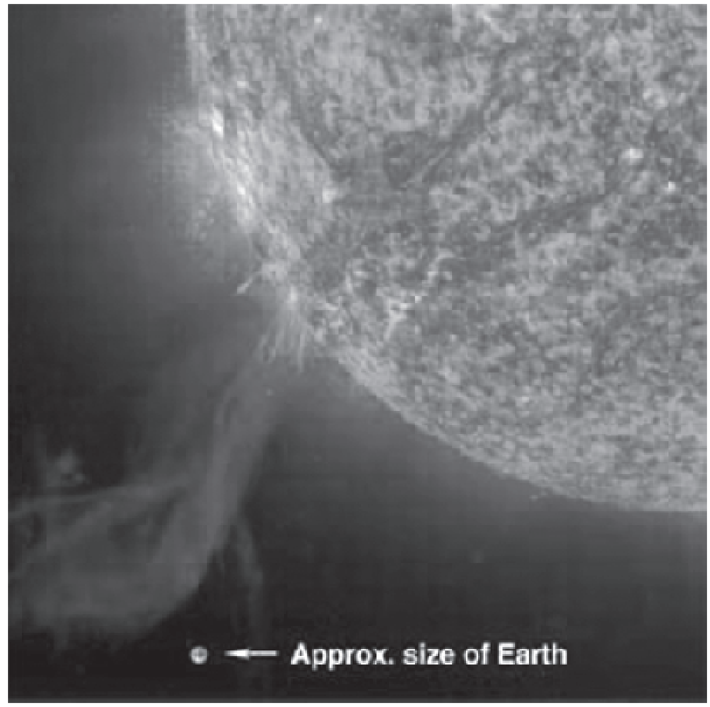



**Figure 2 — A large solar flare, as shown at www.suntrek.org/images/big-solar-flare.jpg.**

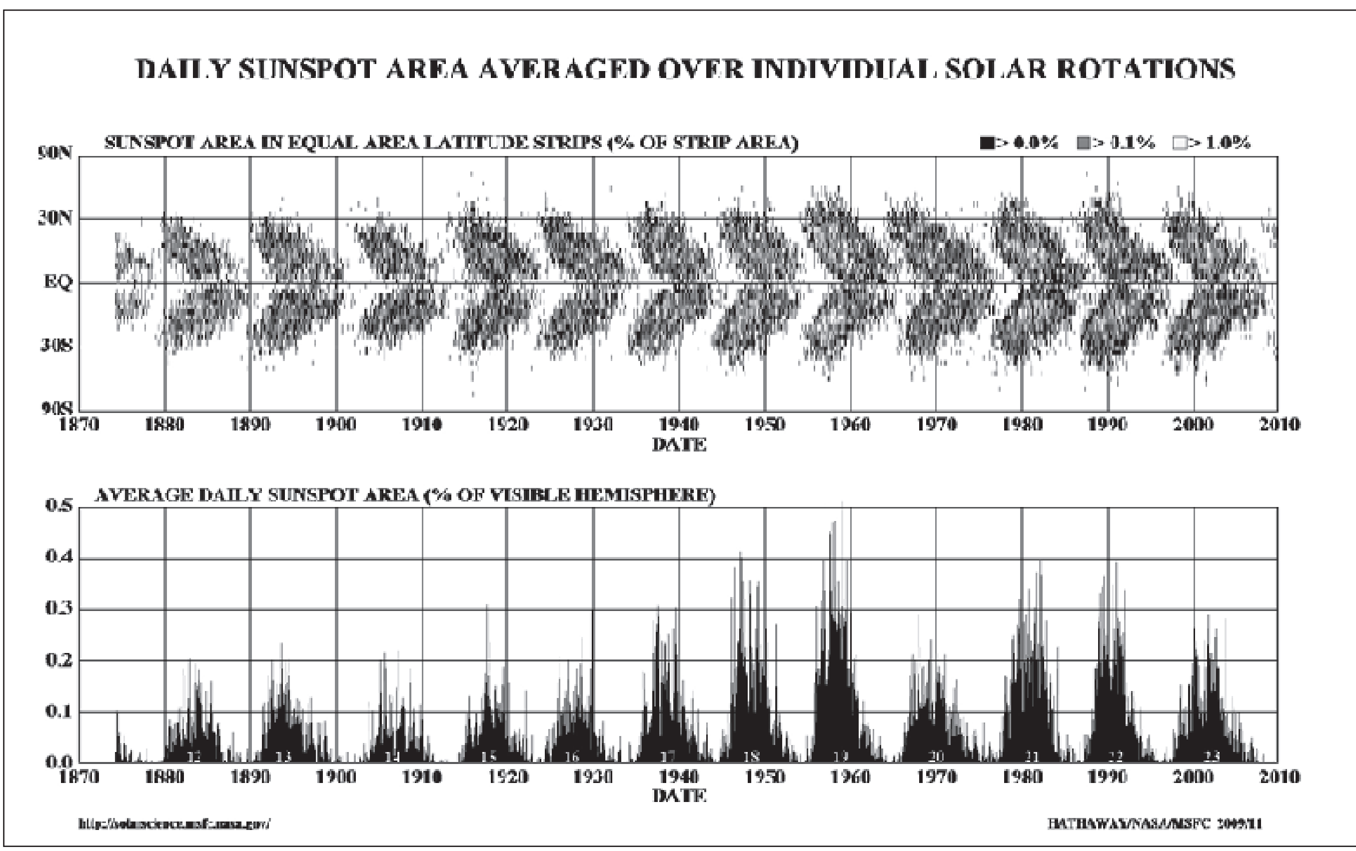



**Figure 3 — This graph of multiple sunpot cycles shows how the positions of the sunspots move from the polar regions towards the equator as the cycle progresses. This creates a butterfly pattern. The image is from http://upload.wikimedia.org/wikipedia/commons/9/93/Sunspot_butterfly_with_graph.jpg.**

the height of the D region. (Remember that $\lambda = c / f$, thus 300,000 (km/s) / 20,000 Hz = 15 km). In addition, the signal refracts in the D region now, and less loss is experienced since it no longer passes through the D region to refract in the E region. This generally leads to a sudden increase in VLF signal, called SID (Sudden Ionospheric Disturbance). Sometimes the VLF signal could be reduced (as with my particular VLF radio) because the low refractions have more collisions of waves and this leads to increased destructive interference.

A quick way to check the performance of your VLF radio is to monitor sunrise and/or sunset. Remembering that the D region is dependent on solar energy, our radios can detect the changes in the D region at both times of day as the sun passes over the transmitter/receiver path. There is generally a small peak produced at both points on the charts (see Figure 11 for a clear example). The effect lengthens with an increase in distance (east/west) between the transmitter and receiver.

**X-Ray Flare Classes**

Rank of a flare based on its X-ray energy output. Flares are classified by the order of magnitude of the peak burst intensity (I) measured at the earth in the 1 to 8 angstrom band as follows:

| *Class* | *(in Watt/sq meter)* |
|---|---|
| B | $I < 1.0 \times 10^{-6}$ |
| C | $1.0 \times 10^{-6} <= I <= 1.0 \times 10^{-5}$ |
| M | $1.0 \times 10^{-5} <= I <= 1.0 \times 10^{-4}$ |
| X | $I >= 1.0 \times 10^{-4}$ |

**Figure 4 — Solar flares are classified according to the X-ray energy released in the flare. This chart is from www.swpc.noaa.gov/info/glossary.html#RADIOEMISSION.**

## The Radios

Mark Spencer spent a lot of time in a Sep/Oct 2008 *QEX* article describing a VLF radio system, which he designed and built, so I will merely provide some basic information about my system here.[1] The radio I chose uses a design by the American Association of Variable Star Observers (AAVSO). Since the sun is a variable star and easy to observe, it is an ideal source. I use the 24.0 KHz signal from Cutler, Maine, but there are signals throughout the country you can monitor. Check out the AAVSO Web site for all the relevant information and schematics: **www.aavso.org/observing/programs/solar/sid.shtml** (follow the links on this page to equipment and data recording).

I built the Gyrator II radio using a circuit board from Far Circuits (**www.farcircuits.net/receiver1.htm**) made for this device, making it much easier to build. Most parts are available from RadioShack and are simply positioned on the circuit board and soldered into place. Figure 5 is the schematic diagram of the Gyrator II VLF receiver. I

[1]Mark Spencer, WA8SME, "SID: Study Cycle 24, Don't Just Use It," *QEX*, Sep/Oct 2008, pp 3-9.

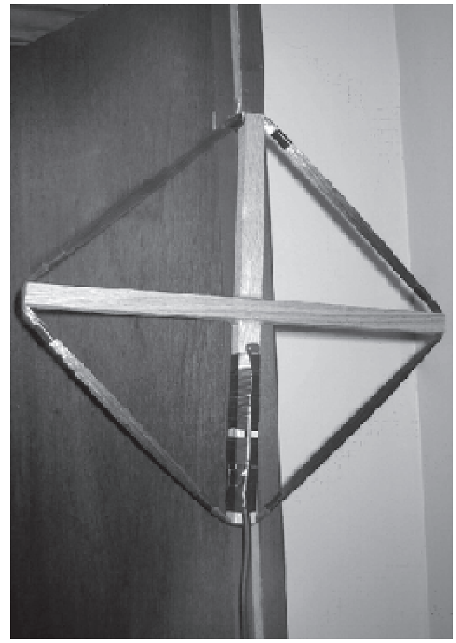


**Figure 6 — Here is a photo of my loop antenna.**

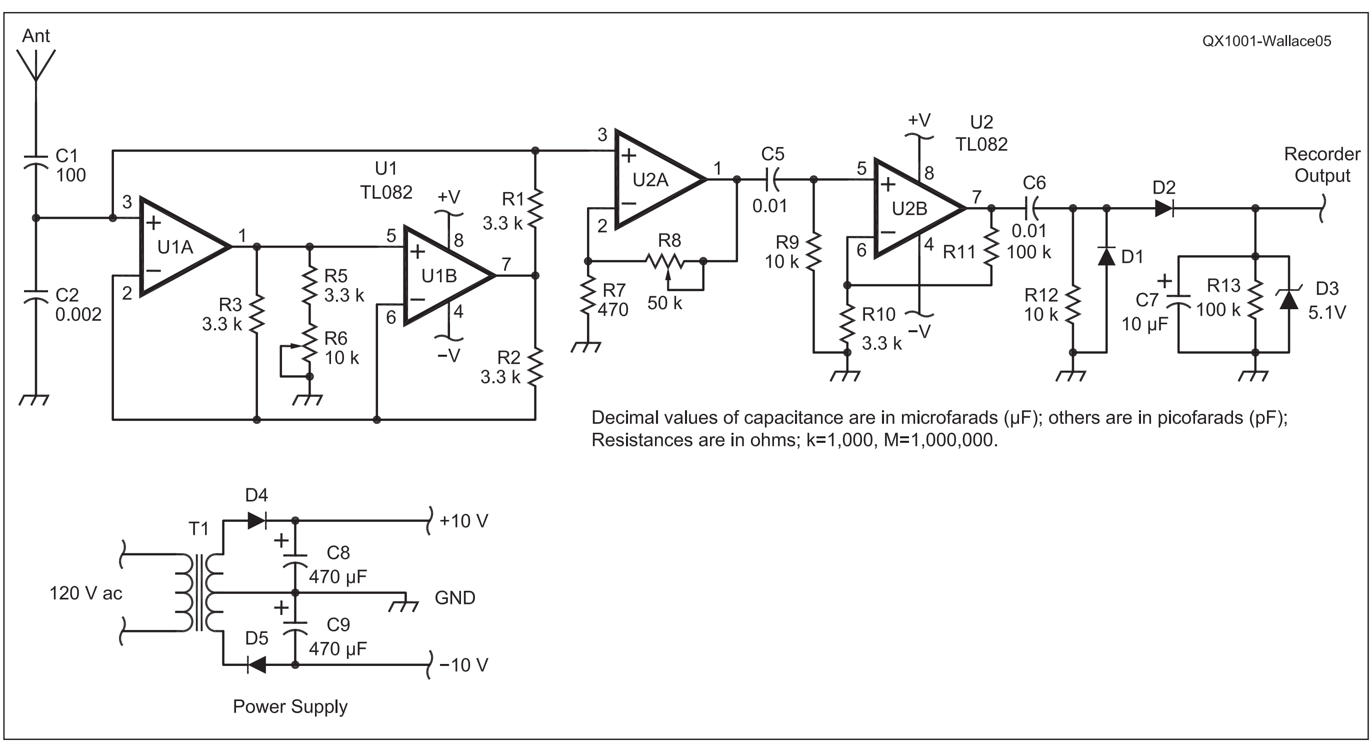



**Figure 5 — This schematic diagram shows the Gyrator II VLF receiver, with parts list. There is more information at www.aavso.org/images/fullgyrator.gif.**

Figure 7 — This photo shows my complete system, with the loop antenna at top left, a computer on a shelf and the VLF radio and analog to digital converter device in a box below the computer.

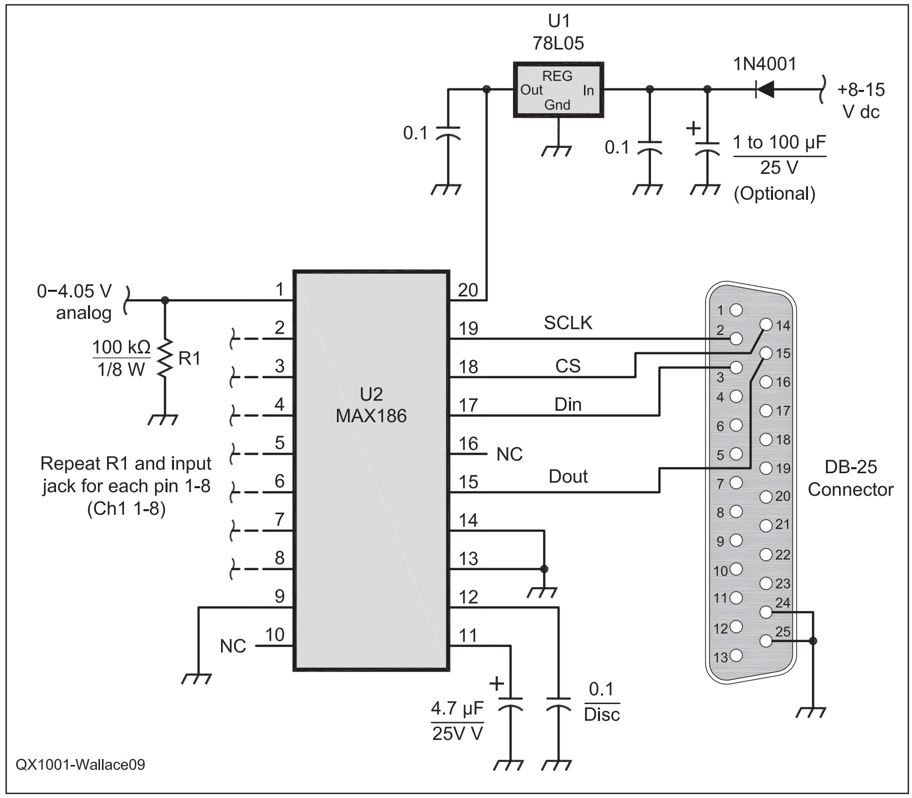



Figure 9 — This schematic is the ADC device from Jim Sky at his Radio Sky Web site: www.radiosky.com/skypipehelp/skypipe8channelADC.html.

Figure 8 — Here is a photo of the antenna field for station NAA in Cutler, Maine.

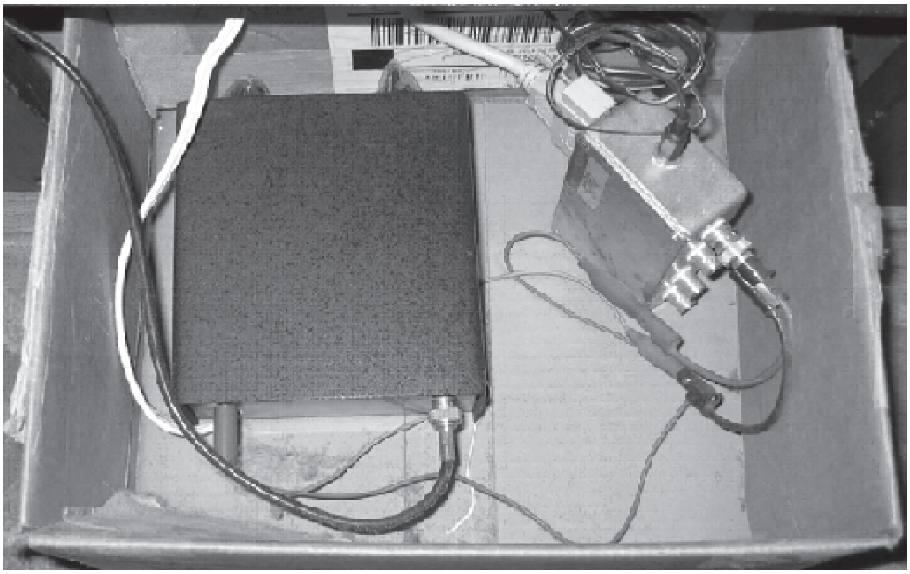


**Figure 10 — Here is a close-up photo with my VLF radio on the left and the ADC device on the right.**

## Stanford Radio - By Bill and Melinda Lord (SARA)

Tim Huynh of Stanford University has designed a simple system to monitor Sudden Ionospheric Disturbances (SID). He uses a preamp that feeds a signal into a sound card, which records at up to 96 kHz to collect data at very low frequencies. The preamp is connected to a one meter loop antenna made of 400 feet of solid copper wire. The software can be easily configured to monitor multiple frequencies. Commonly monitored frequencies used in the US range from 21.4 kHz to 25.2 kHz. The unit can monitor from 7.5 kHz to 43.7 kHz.

Since the system can collect several frequencies, the antenna is not tuned. The early designs required an 8 foot loop antenna made of 720 feet of solid copper wire, but the current design works with a compact one meter antenna. You can increase the sensitivity with a larger antenna if you wish. Stanford is producing 100 units with the assistance of the Society of Amateur Radio Astronomers (SARA) and will be distributing them to schools all over the world.

used an IC socket so I wouldn't be soldering the IC directly; construction was very fast and easy. I used an antenna design from Mike Hill — "A Tuned Indoor Loop." See Figure 6. It is easy to make but consists of 125 turns of magnet wire! Keeping count can be a challenge. The wire is wound around a small frame (I made mine out of wood). This is an easy project, well within the skills of any ham. Figure 7 shows my complete receiving system.

Another very interesting monitor is available from Stanford University for about $270 (**nova.stanford.edu/~vlf/IHY_Test/pmwiki/pmwiki.php?n=Main.HomePage**). It uses multiple frequencies to monitor flare activity. I have included a short summary at the end of this article.

### Tuning

I found that using a coil of wire attached to a signal generator set to the station frequency (24.0 kHz for Cutler Maine) provided a great tuning device. Simply turning the potentiometer for gain and then tuning the radio, searching for the peak voltage output, made short work of it.

### NAA

The NAA transmitter is located in Cutler Maine, and is maintained by the Navy for communication with submarines. It operates at 24.0 kHz and has a sister station at 24.8 kHz called NLK in Jim Creek, Washington. The NAA station occupies 3,000 acres and used 90,000 cubic yards of concrete and 15,000 tons of steel in its construction. It produces a 2 MW signal. The antenna consists of 13 towers, the center at

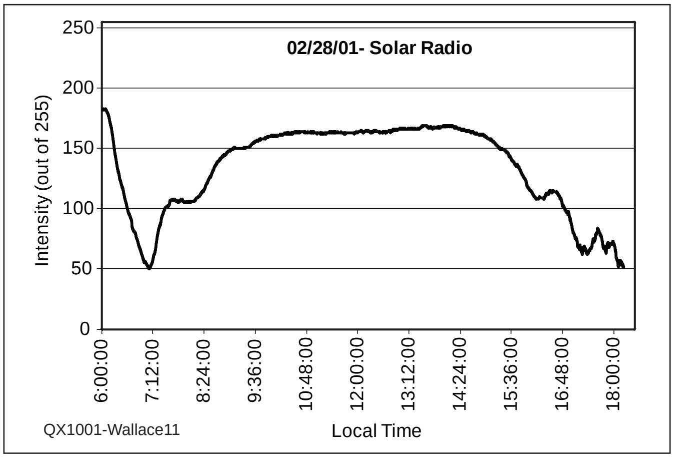



**Figure 11 — This graph represents data collected on a quiet day, when no flares occurred.**

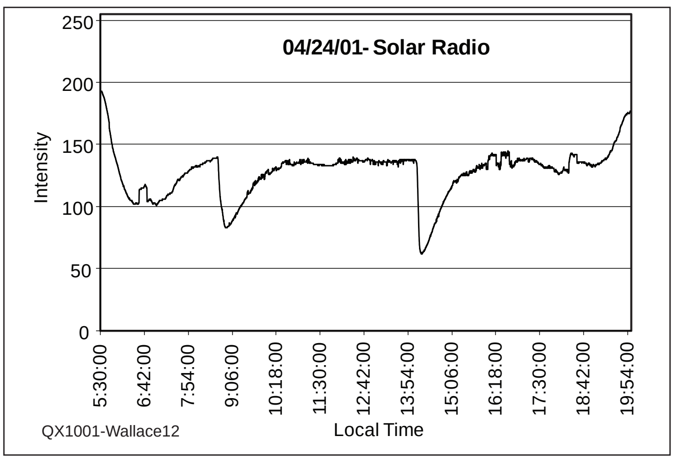



**Figure 12 — This graph shows a day with several large solar flares. Also notice the peaks at both ends of the graph, representing the sunrise/sunset effect.**

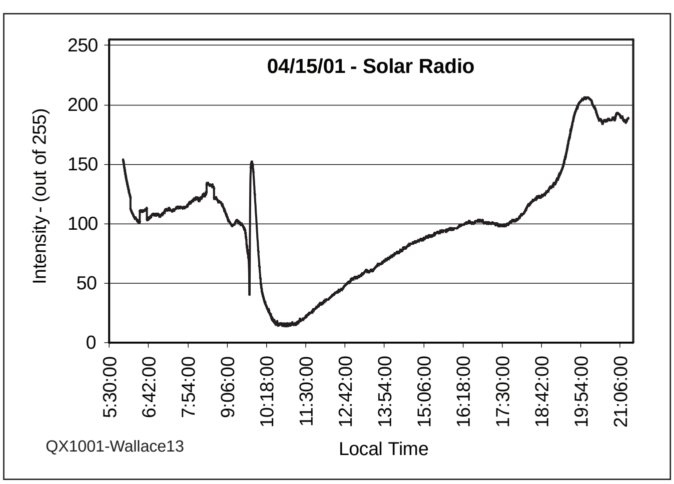



**Figure 13 — This graph is from a day that had an amazing X 14.4 flare.**

980 feet above ground, the next six at 875 feet and the last six at 800 feet. The antenna consists of 75 miles of 1 inch phosphor bronze wire above ground. See Figure 8 for a picture I took on a recent visit.

### Data Recording/Charts

I record my data through an analog to digital converter (ADC) I built using plans at Jim Sky's Web site (**www.radiosky.com/skypipehelp/skypipe8channelADC.html**). The schematic is shown in Figure 9. It is a simple device that allows you to gather data through the printer port of any computer using Jim's software called *Radio-SkyPipe*. Figure 10 is a photo of my receiver and the ADC in separate project boxes. I use an old laptop with *Windows* 3.1. I then export the data to floppy disk and bring it to my main computer and make the final charts in Microsoft *Excel*. In Figure 11, a quiet day with no flares present is shown. Note the peaks on both ends — the sunrise/sunset effect. Figure 12 shows a day with several large flares. Flares usually appear as upward peaks, but on my receiver they appear as downward troughs. These are M1.6 and M2.3 flares respectively. Figure 13 shows a day with an amazing X 14.4 flare! Data can be reported each month to AAVSO through a simple log program, in which you enter the date and begin, peak and end times of each flare.

## Conclusion

I hope you will try this simple radio and send your results to AAVSO. If you have any questions, comments or concerns, feel free to contact me.

*Jon Wallace has been a high school science teacher for over 28 years. He is past president of the Connecticut Association of Physics Teachers and was an instructor in Wesleyan University's Project ASTRO program. He has managed the Naugatuck Valley Community College observatory and run many astronomy classes and training sessions throughout Connecticut. Jon has had an interest in "non-visual" astronomy for over twenty-five years and has built or purchased various receivers as well as building over 30 demonstration devices for class use and public displays. He is currently on the Board of the Society of Amateur Radio Astronomers (SARA) and developed teaching materials for SARA and the National Radio Astronomy Observatory (NRAO) for use with their Itty-Bitty radio Telescope (IBT) educational project. Other interests include collecting meteorites, raising arthropods ("bugs") and insectivorous plants. Jon has a BS in Geology from the University of Connecticut; a Master's Degree in Environmental Education from Southern Connecticut State University and a Certificate of Advanced Study (Sixth Year) in Science from Wesleyan University. He has been a member of ARRL for many years but is not a licensed Amateur Radio operator.*

For further information check out:

**Society of Amateur Radio Astronomers (SARA)** — **www.radio-astronomy.org/**

**AAVSO** — **www.aavso.org/observing/programs/solar/sid.shtml**

**Stanford's SID Program** — **solar-center.stanford.edu/SID/sidmonitor/**

Ian Poole, G3YWX, ***Radio Propagation — Principles & Practice***, available from your local ARRL dealer or from the ARRL Bookstore. Telephone toll free in the US: 888-277-5289, or call 860-594-0355, fax 860-594-0303; **www.arrl.org/shop**; **pubsales@arrl.org**.

QEX

**Cadet Thomas Dean, KB1JIJ**

5 Mohegan Rd, Groton, CT 06340; thomas.dean@usma.edu

# A Digital Frequency Hopping Spread Spectrum Transmitter

*What can you program an FPGA to do? A digital logic class project leads to this FHSS transmitter.*

Software-defined radio is quickly transforming the communications field. Software-defined radio uses software to perform functions such as modulation, demodulation and filtering, all in the digital realm. The advantage to this is that it allows for systems to be built with fewer components than a traditional analog radio architecture, and it allows for an incredible amount of flexibility within the system, even after it has already been produced. There are several common approaches to implementing a software-defined system. One way that most hams may be familiar with can be done on a computer and converted to or from the analog realm through either a sound card or a peripheral such as a Universal Software Radio Peripheral. It is also possible to do the processing on a programmable chip such as a field programmable gate array (FPGA) or through the use of an application specific integrated circuit (ASIC), microprocessors, or DSP chips. In this article I would like to present an example of a project that creates waveforms by using only two components: an FPGA and a video digital to analog converter (DAC). This article will also present an introduction to spread spectrum (SS) systems, which are not commonly seen on the amateur bands.

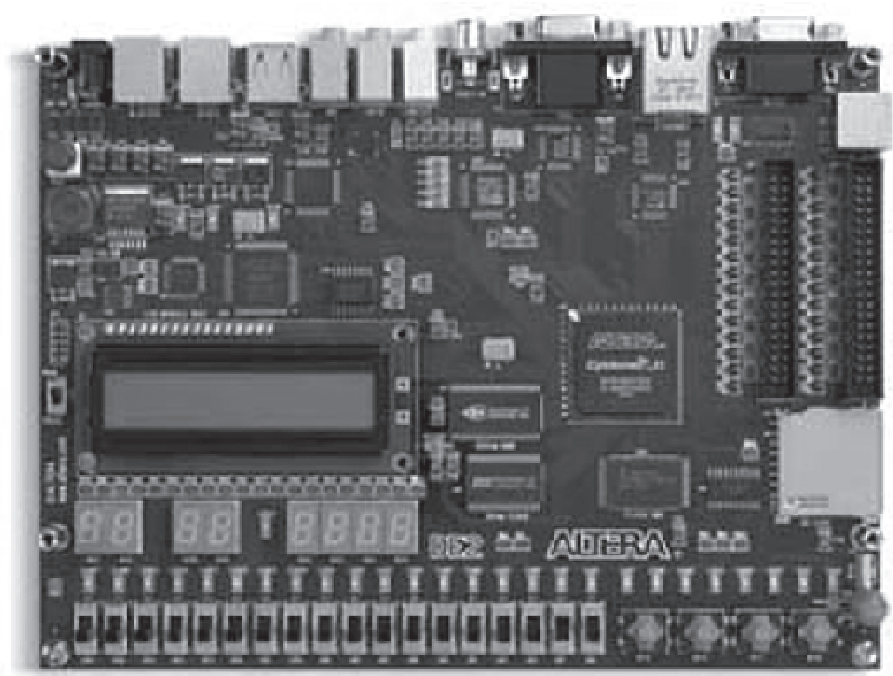


**Figure 1 — The Altera DE2 Development and Education Board.**

## Introduction

This was a project that I designed for a digital logic class. The project was required to be designed on the Altera DE2 Development and Education board, using an Altera Cyclone II FPGA. The DE2 board is designed to allow students to gain experience programming digital systems, minimizing the hassle of fabrication. The board is also useful for prototyping systems before production. This board contains many built in I/O devices and interfaces, as well as on board memory and displays. See Figure 1. For my project, I chose to design a software-defined frequency hopping spread spectrum transmitter. The output of the board contains a signal in the range of 24 kHz to 6.2 MHz. This signal would have to be converted to an amateur band to be used on the air by an Amateur Radio operator. That is an operation that is possible, but somewhat difficult to do. Rather than being presented as a fully working project, this paper presents a demonstration of the sort of system that could be contained in a single chip and how it was created.

The Cyclone II FPGA generates samples of the waveform and then sends them to an ADV7123 Digital to Analog Converter (DAC) to create the IF output of the transmitter. The data rate, frequency range and hopping rate of the transmitter are easily adjusted within the source code. It would not be too difficult to program an interface to allow the user to configure these values. I used a 100 kilobit per second data rate with a 400 kHz hopping rate. There are 16 possible code division multiple access (CDMA) patterns associated with this design, each

[1]Notes appear on page 16

of which has 255 channels that are spaced approximately 24 kHz apart. The total bandwidth of the transmitter is 6.25 MHz.

I developed the transmitter portion of this project, while my project partner, Cadet Adam Royal worked on interfacing it with a hex keypad over the expansion header and the 2 × 16 digit LCD display on the DE2 board. The keypad allowed selection of the CDMA sequence and the LCD displayed the sequence number. In this article, I will present the design and theory behind the transmitter.

Frequency hopping spread spectrum (FHSS) was chosen in part due to the relative simplicity of creating the waveform. In addition, I chose a spread spectrum mode to help me gain a better understanding of spread spectrum systems, which are of great importance in military and many commercial communication systems.

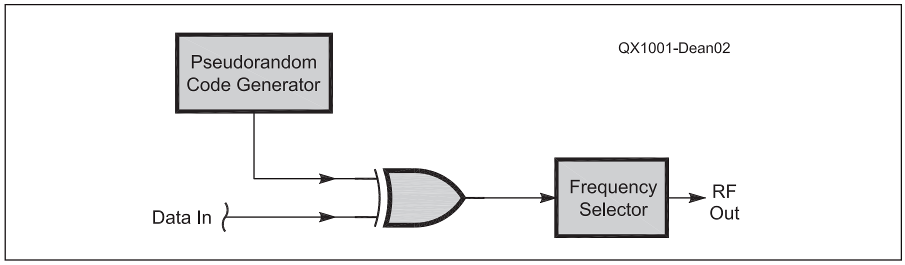



**Figure 2 — Here is a simple block diagram of a frequency hopping spread spectrum (FHSS) system.**

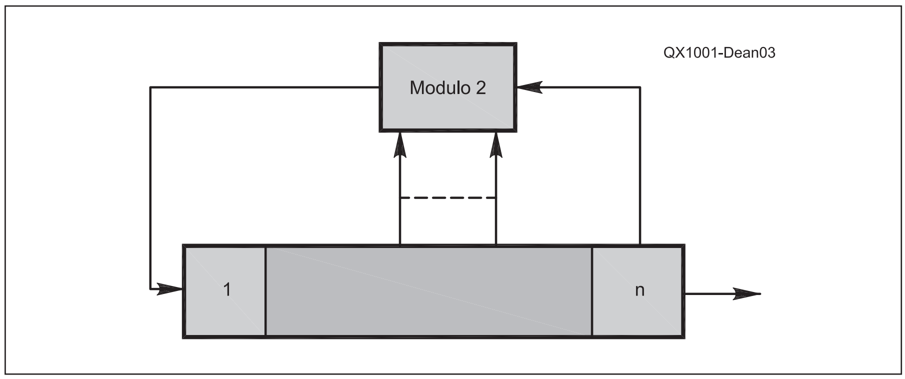



**Figure 3 — This is a simple block diagram of a linear sequence generator.**

| Sequence Number: | Tap positions: |
|---|---|
| 1 | 8,4,3,2 |
| 2 | 8,6,5,3 |
| 3 | 8,6,5,2 |
| 4 | 8,5,3,1 |
| 5 | 8,6,5,1 |
| 6 | 8,7,6,1 |
| 7 | 8,7,6,5,2,1 |
| 8 | 8,6,4,3,2,1 |

**Figure 4 — This table shows the tap positions for maximal codes.**

## Spread Spectrum Systems

Spread spectrum emissions are not commonly seen on the amateur bands; however, they offer several advantages over traditional narrowband modes. From the point of view of military communications, they make it much more difficult for a signal to be jammed. For example, consider a typical 5 kHz FM voice signal. This could be jammed by simply transmitting on top of it. If the signal was instead hopped over a bandwidth of 30 MHz, it would become nearly impossible to jam unless the jammer had knowledge of the spreading pattern. Commonly used spread spectrum modes in the military include the Single Channel Ground-Airborne Radio System (SINCGARS) and HAVEQUICK system. The use of a spread spectrum system for this purpose is often referred to as an electronic counter-counter measure (ECCM).

Spread spectrum systems are also of great advantage outside of the military. They offer resilience to fading and can often be hidden below the noise floor. By spreading a signal across a large portion of the spectrum, a communications system will have what is known as a process gain. Process gain is the ratio of the spread signal bandwidth to the unspread (baseband) signal bandwidth, and is usually expressed in decibels. For example, if a 1 kHz signal is spread over a 100 kHz bandwidth, the ratio will be 100, or a 20 dB process gain. (For more information on process gain, see **http://en.wikipedia.org/wiki/Process_gain**.)

Depending on the amount of spreading, a system can also gain additional rejection of interference. SS systems can offer a way for multiple users to send data over the same set of frequencies, even though it is a large frequency range. Users must simply choose spreading codes that have low levels of correlation to each other. This is referred to as code division multiple access (CDMA). Most protocols under IEEE 802.11 use spread spectrum systems to allow multiple users on the same frequency range. In addition, the Bluetooth protocol as well as many cellular telephone networks and GPS systems employ a spread spectrum modulation method.

## Programming FGPAs

Before I get into the details of the transmitter design, I will introduce the main component of the design — the FPGA — for those who aren't familiar with it. Programming an FGPA is slightly different from writing code for a computer or microprocessor. An FPGA is a very large array of logic cells. Each cell contains a number of inputs and outputs, and can be programmed to perform a certain logic function. Cells can also function as counters, adders, or in some cases multipliers. Most FPGAs also have built in SDRAM and sometimes incorporate PLLs or DSP units into their design. When you write a program for an FPGA, rather than defining a set of instructions for a processor to follow, you are describing how the cells within the FPGA are connected.

FPGAs are programmed with a hardware description language (HDL). This project was coded using *VHDL* (Very High Speed Integrated Circuit HDL), which is similar in syntax to *ADA*. Programming an FPGA can be very tricky at first. This may be especially true for someone who is familiar with computer programming. The main difference is that your code is executed in parallel rather than sequentially. Once the code is written, it can be synthesized to the Register Transfer Logic (RTL) which is a description of how registers and logic elements are connected. From this, the circuit's actual wiring can be derived. This is useful for purposes of simulation, which can be used to determine if your circuit will execute properly before it is programmed. From here, the circuit is fitted to the chip and complied into a line code that can be programmed on the chip. This is generally done using a program that is provided by the chip manufacturer. The software package used in this project is *Quartus II*, a trial version of this package is freely

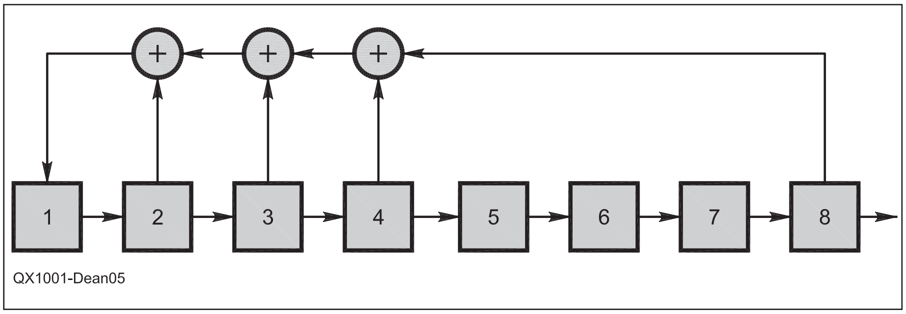



**Figure 5 — This block diagram shows an 8-bit maximal linear code generator. The circles with a plus inside represent 1 bit adders (with no carry out), but can also be considered as exclusive OR (XOR) gates.**

| | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 |
|---|---|---|---|---|---|---|---|---|
| 1 | - | 6 | 7 | 6 | 5 | 4 | 4 | 5 |
| 2 | 6 | - | 13 | 6 | 10 | 4 | 4 | 7 |
| 3 | 7 | 13 | - | 5 | 9 | 4 | 4 | 6 |
| 4 | 6 | 6 | 5 | - | 8 | 6 | 5 | 6 |
| 5 | 5 | 10 | 9 | 8 | - | 8 | 5 | 10 |
| 6 | 4 | 4 | 4 | 6 | 8 | - | 7 | 5 |
| 7 | 4 | 4 | 4 | 5 | 5 | 7 | - | 6 |
| 8 | 5 | 7 | 6 | 6 | 10 | 5 | 6 | - |

**Figure 6 — This table gives the cross-correlation values for 8-bit sequences.**

available from Altera. The DE2 board is programmed from a computer USB port using a JTAG interface. The programming files can also be stored on non-volatile EEPROM, which is loaded onto the FPGA each time it is powered up.

## Design Theory

### Frequency Hopping Spread Spectrum Communications

There are many different methods for spreading data across a wide bandwidth.[1] Two of the most common methods are direct sequence spread spectrum (DSSS) and frequency hopping spread spectrum (FHSS). DSSS takes a signal and multiplies it by a pseudorandom noise signal. As a result, the signal resembles white noise and simultaneously occupies a large bandwidth. In contrast, FHSS systems use a pseudorandom sequence (known to both the transmitter and receiver) to change the carrier frequency.[2] In my transmitter, the frequency changes at a rate faster than the data transmission. FHSS is essentially a form of frequency shift keying.

The data is added to the pseudorandom sequence by an exclusive OR (XOR) digital logic operation. Then the data can be recovered by XORing the sequence again, since $A \oplus B \oplus A = B$. A block diagram of a typical FHSS transmitter is shown in Figure 2.

### Generating Pseudonoise

One component that is critical to designing a spread spectrum system is the pseudorandom code generator. The code that is generated must be deterministic so that it can be decoded by the receiver. Generally, the use of a spread spectrum signal is not to protect against eavesdropping; this is accomplished more effectively through encryption. The two important features that must be considered in the code are called cross-correlation and autocorrelation.

Autocorrelation is a measure of similarity between the code and a phase-shifted version of itself. It can be defined as the number of times that the values of a phase shifted version of the code are equal to the values of the original code. This property is of extreme importance to our system, due to its coherence. For a signal to be decoded, the receiver must be able to synchronize its code sequence to the transmitted code sequence (a process referred to as correlation). If a sequence has a high autocorrelation then it may become difficult to determine the correct point to correlate the codes.

Another important property of the code is cross-correlation. This property is very similar to autocorrelation, but measures the amount of correlation of one sequence to a different sequence. This property is important to our system if it is to allow for CDMA. Two communicators may share a single set of frequencies only if there is a low level of correlation between the set of codes that they use.

I first chose a linear code for our system. A linear code can be generated by a shift register with a configuration similar to the one shown in Figure 3.

Consider a code that is generated by a sequence generator that has n stages, then it would have a length of $2^n–1$ (a value of zero is not possible). Dixon defines a code to be a "maximal sequence" if the following properties apply to this code:[3]

1) Every possible state exists at some point.

2.) No values repeat themselves within one period of length $2^n–1$.

3.) The number of ones in a sequence equals the number of zeros in the sequence plus one.

4.) The statistical distribution of ones and zeros is well defined and always the same.

A maximal code would be the ideal code to use for our system since it has the lowest possible autocorrelation and is easy to generate. Generally, the larger the number of states in a system, the better. It will be difficult, however, to generate a waveform at a precise frequency using strictly digital logic. Therefore, it is best not to choose too long of a sequence. A length of eight bits seemed like a reasonable medium. There are sixteen possible sets of maximal codes that can be made from an eight stage generator. The table in Figure 4 shows half of these possible codes, the remaining eight are the mirror images of these codes.[4] Figure 5 will clarify how these sequences are defined.

This generator is a shift register that feeds back into itself. With each clock cycle, the bits are shifted to the right one register. The register is tapped at various locations as shown, which allows the values on the register to be changed in a cyclic process with a period of 255 clock cycles. Since this code will be used to select frequencies, we will want the output to be in parallel. Using the values from each register at each clock cycle, we will end up with 255 states, corresponding to 255 different frequencies.

Each of the sixteen different possible sequences will correspond to different channels that can be used in the CDMA scheme. We are therefore interested in the cross-correlation between each sequence, as this will show how often the channels will interfere with each other.

The table in Figure 6 shows an index of cross-correlation between half of the codes, generated in-phase with each other. It was calculated by generating every possible code in a Microsoft *Excel* spreadsheet, and then comparing the values of the codes. The index represents the number of times that the two codes shared the same value. With the exception of approximately five occurrences, this only occurred at the beginning and ending of the codes.

This data only shows a very small portion of the possible cross-correlation values since it is very rare that two codes will be exactly in phase. Studies have determined that a maximal sequence of length eight will overlap the remaining fifteen other maximal sequences between 31 and 95 times.[5] While this may not result in any gain in spectral efficiency, it would still be possible to achieve error free communication using this scheme with sufficient forward error correction.

### Convolutional Coding

The purpose of forward error correction (FEC) is to add a level of redundancy to a transmission to allow the receiver to correct some errors.[6] These errors can come from many sources including thermal noise, jammers or communicators on other channels. There are many ways to implement this redundancy. One form of forward error correction that is very simple to implement and provides us with a relatively high coding gain is convolutional coding. While this code is very simple to encode, it is rather complicated to decode. This can be accomplished through what is known as a Viterbiti decoder;[7] however, that is beyond the scope of this article since this project simply focuses on the transmitter. Convolutional codes are usually specified by three parameters:

$n$ = number of output bits
$k$ = number of input bits
$m$ = number of registers

Not all forms of convolutional code create good codes. One commonly used form of this code[8] is (2,1,7), with tap positions defined as:

P1: 1111001
P2: 1011011

This can be done with the encoder shown in Figure 7.

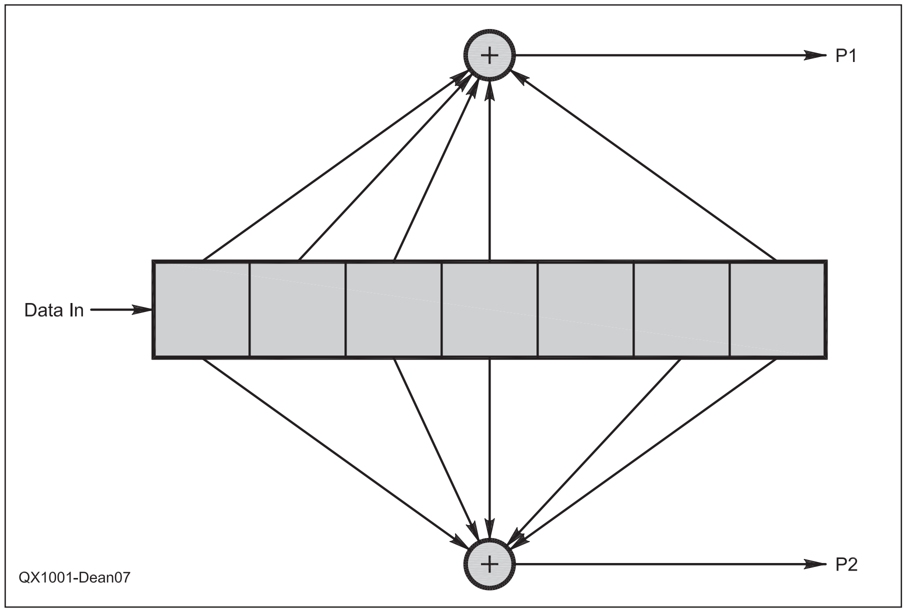



**Figure 7 — Here is a block diagram of a convolutional encoder.**

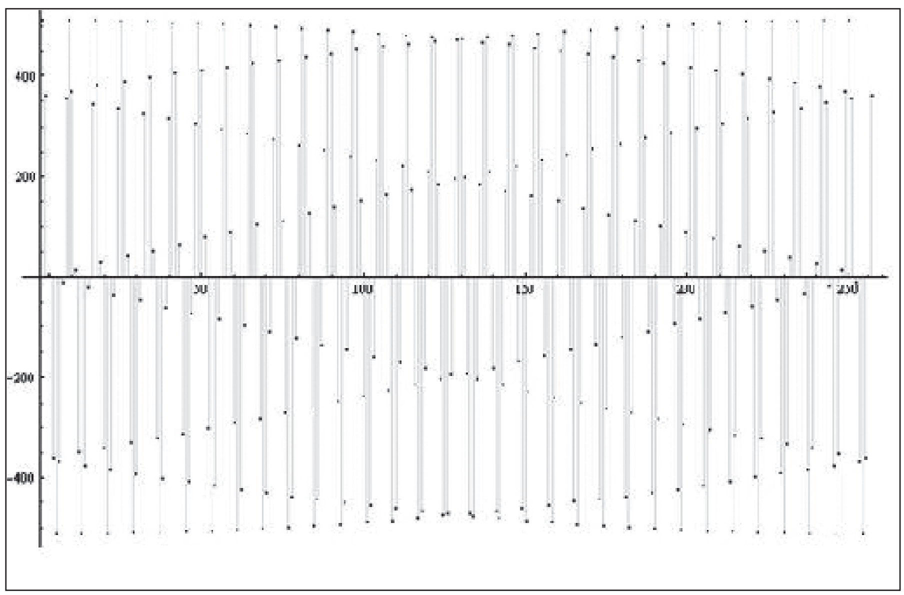



**Figure 8 — This waveform results from sampling the look up table once every 255 values.**

This encoder is a shift register, similar to the pseudonoise generator described earlier. Instead of XORing bits and feeding them back into the register, they are instead passed as two separate lines. This provides an amount of redundancy within the system. In order to use the output of this encoder, it is converted into an 8 bit parallel line at a lower clock rate, where it can be XORed with the pseudonoise bit for bit.

This simple coding technique can allow signals that are up to 5 dB weaker to be received with the same error rate.[9] The disadvantage to using this coding technique is that it doubles the amount of data that is sent. This means that twice the energy is being sent by the radio, resulting in a coding gain of 2 dB (5 dB – 3 dB). It is also highly effective to add Reed-Solomon coding before performing convolutional coding. Since Reed-Solomon coding is a form of block error correction, these two forms of error correction complement each other very well and result in a significant code gain. Reed-Solomon coding works by creating a polynomial out of the data and then adding redundancy by oversampling the polynomial. Creating a Reed-Solomon encoder on the register level is not as hard as it might seem, but the theory behind how it is done lies in abstract algebra. These two forms of error correction are used together in most deep space and satellite communications.[10]

### Generating the waveform

After the pseudorandom code is modified by the data, we are left with a series of eight bit values, which can be directly corresponded with a frequency. Determining the best way to digitally synthesize a wide range of frequencies was perhaps the most difficult aspect of this project. The initial method that I investigated involved generating clock signals and then converting these to a sinusoidal wave through tunable digital filters. This method would be complex from the design perspective, but very simple when it actually came to implementing in hardware. This method has been used in the design of wideband CDMA systems before, and is referred to as pulse shaping.[11] Eventually I decided not to use this method, and chose to use a

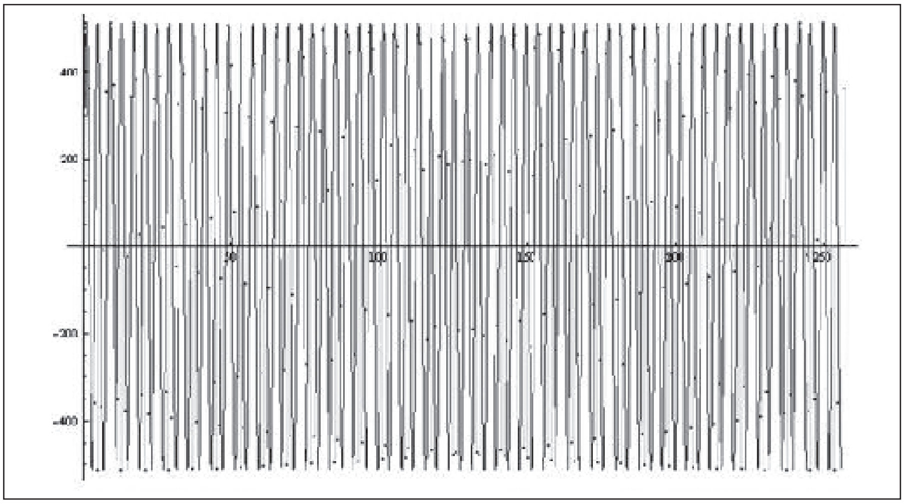



**Figure 9 — Here, the signal resulting from sampling the look up table every 255 values is superimposed on the ideal waveform.**

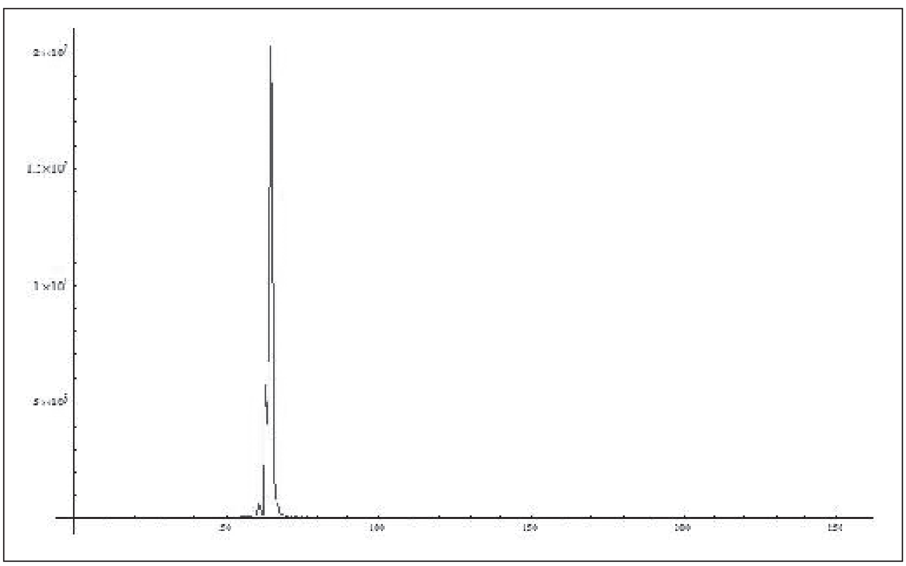


**Figure 10 — This is the spectral display of the waveform that results from sampling the look up table once every 255 values.**

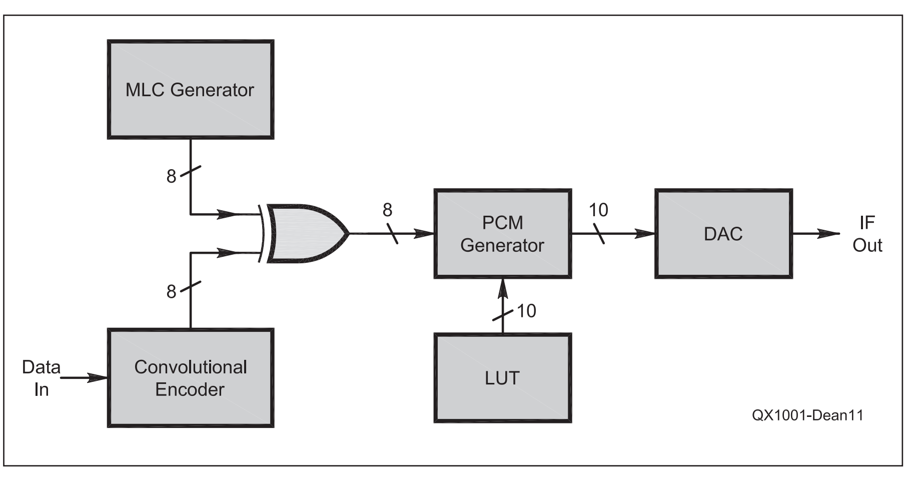



**Figure 11 — This is the final block diagram of the transmitter.**

large look-up table to generate the wave, once I discovered how easily both *Quartus II* and the Cyclone II handle internal memory.

The DAC that is on the Altera DE2 board uses a process known as zero-order hold to convert a set of digital samples to a piecewise analog function. For this project, samples will be sent to the DAC at a rate of 50 MHz. Each sample will be a ten bit value that corresponds to the amplitude of the resulting waveform. Sine waves will be generated by taking values from a look-up table. To change the frequency, I change the number of times that the table is sampled. For example, if every other value of the table is taken, then the resulting waveform will be twice the frequency if all values had been sampled.

### Look-up Table

The look-up table must have, at a minimum, twice the number of possible frequencies because of the Nyquist criteria. Since there are 256 possible values for the frequency, there must be a minimum of 512 values in the table. Since storage inside the chip is not an issue, I chose to double this to improve the quality of the waveform generated. Then, because of the symmetry of the sine wave, the number of entries can be reduced to a quarter of a sine wave. This results in 256 values in the table. The word size within the look-up table must be equal to 10, since we are using a 10-bit DAC. The final amount of memory needed is $2^9 \times 10$ or 5,120 bits. A *Perl* script was found on-line to automate the process of making this look-up table.[12] It came with VHDL code that could be used to access the table to generate a sine wave. The code had to be modified in order to allow for the sampling interval of the table to be modified with input. In addition, the code used two's-complement in order to make the values of the wave signed. The code had to be modified in order to make the values unsigned.

### Spectra

In order to determine the frequencies that result from sampling our table we must consider that the table will be accessed at a rate of 50 MHz or $5\times10^7$ times per second. If $n$ is equal to the output of the modified pseudonoise generator (a value from 1 to 255), then the resulting frequency will be:

$$f(n) = 5\times10^7 \frac{n}{2048} \quad \text{[Eq 1]}$$

where 2048 is equal to the period of the table. This means that the system will have outputs between 24.41 kHz and 6.225 MHz.

When we sample the table at intervals that are not factors of 2048, the resulting waveform will not be purely sinusoidal. Consider when the table is sampled at an interval of 255. The resulting signal can be expressed as:

$$e^{\frac{-255\pi\ in}{1024}} \quad \text{[Eq 2]}$$

where $n$ is a positive integer. Figure 8 displays the waveform graphically. Figure 9 compares this signal to the desired waveform, Sin (255 $\pi$ $n$ / 1024).

It is evident that the waveform is simply not sampled at values corresponding to peaks of the wave. To ensure that this imperfection has no negative impact on the system, it can be converted into the frequency domain using a discrete Fourier transform. The spectrum of the wave is then equal to the square of the transform, which is shown in Figure 10.

The desired signal is about 6 dB higher than any unwanted products. While this is not ideal, it should be small enough not to cause unwanted effects on the receiver. The waveform generated at this frequency is in fact the worst possible waveform. The amplitude of the output signal in Figure 8 appears to show a low beat frequency. Normally, the output signal would be put though a reconstruction filter to remove any alias products. If we filtered out any alias products, we would be left with a constant amplitude sine wave.

### Imperfections in the Waveform

One issue that is presented when converting a digital signal to an analog is the result of imaging occurring at higher frequencies, as well as other imperfections that end up in the resulting waveform. Due to its zero-order sample-and-hold operation, a digital to analog converter (DAC) will produce a distortion in the output spectrum that follows the function sin(x) / x.[13] At 80% of the Nyquist frequency, this attenuation will be around –12.6 dB.[14] The maximum frequency reached by the system is 25% of the Nyquist frequency. While this may cause some roll-off, it should not be large enough to have a significant impact on the performance of the system. The roll-off could be countered by using a digital filter prior to the DAC to counter the effects.[15] Another major issue in the resulting signal comes from aliasing that occurs. Replicas of this signal will occur about multiples of the sampling frequency. In some cases, it is possible that these replicas will occur within the radio passband. Thanks to the sample-and-hold nature of the DAC, aliases that fall close to multiples of the sampling frequency will be attenuated; generally, the greater the sampling frequency, the smaller the amplitude of the aliases. One additional form of error that will be present in the waveform will be noise in the output called glitch energy, which is caused by a voltage error in the DAC. Analog smoothing and low pass filters could be used at the output of the DAC in order to improve the final signal. This would help reduce all forms of error mentioned above.

### Interfacing with the DAC

The actual interface to the ADV7134V DAC is very straightforward. All of the wiring was already done on the DE2 board. All that had to be done was to properly select the output pins on the FPGA to drive the chip. The DAC was designed to drive a VGA monitor and therefore has separate DACs on chip. The samples were sent to the red input in this project. In order to get output from the DAC, high signals had to be input to the $\overline{BLANK}$ and $\overline{SYNC}$. In addition, the DAC has an output impedance of 75 Ω. Therefore the signal had to be passed through an RF transformer to match the impedance to the commonly used 50 Ω output. The signal was captured by connecting a VGA cable to the VGA port on the board. I cut off the other end of the cable and installed a BNC connector to the red signal in the cable.

## Final Block Diagram

The block diagram of the final system is shown in Figure 11. The output is intended to be the IF signal of a radio. Bands which would be good choices would include the 70 cm Amateur band and the 2.4 GHz ISM band.

## Simulation and Test Plan

The component that created the digital signal was first simulated on *Active-HDL*, which is an FPGA design and simulation tool. The resulting waveform from the pulse code modulator (PCM) is shown in Figure 12. You can see that the output values increase and decrease in periodic fashion.

Next, the maximum length code (MLC) generator was added to control the input of the PCM generator. Figure 13 shows the waveform from this simulation.

While this waveform seems chaotic,

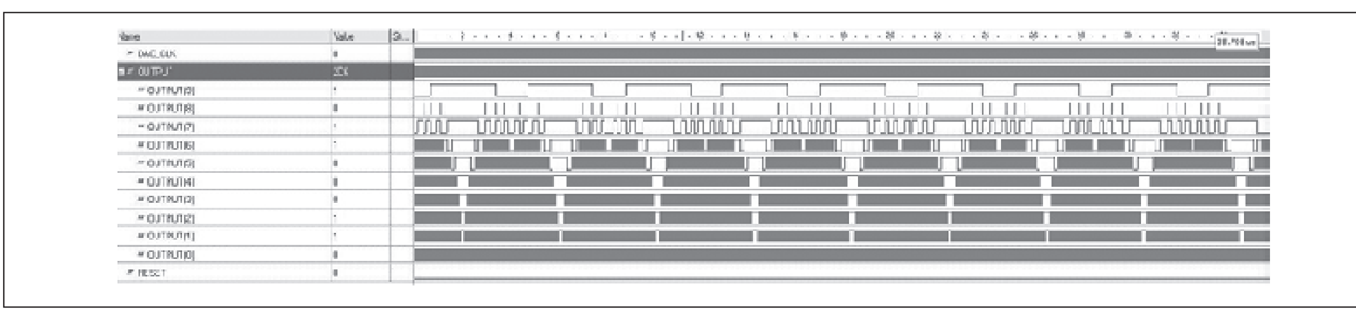


**Figure12 — This display is the output of the pulse code modulator (PCM) generator, as given by the *Active-HDL* simulation.**

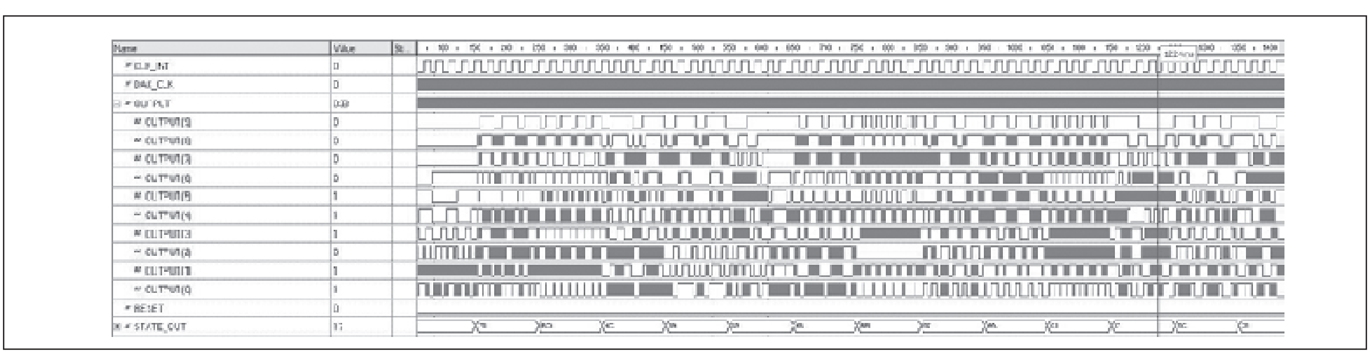


**Figure13 — The *Active-HDL* simulation output of the PCM generator tied to the maximum length code (MLC) generator.**

you can see that it is functioning properly by simply looking at the fourth line on the graph, just below the two black lines (labeled as Output [9] on the left edge of the graph) and the bottom (State Out) line. The fourth line represents the most significant bit of the PCM. It clearly changes its period when the bottom line, which represents the states from the PCM generator, changes state.

The test bench code that was used was identical in both of these simulations. What made them different was that the MLC Generator was connected on the second one.

The next step in the testing process was to test the output of the VGA DAC. This test served to ensure that all components of the radio were working correctly. It is very difficult to test whether or not the forward error correction is working without a working receiver, due to the pseudorandom nature of the transmitter.

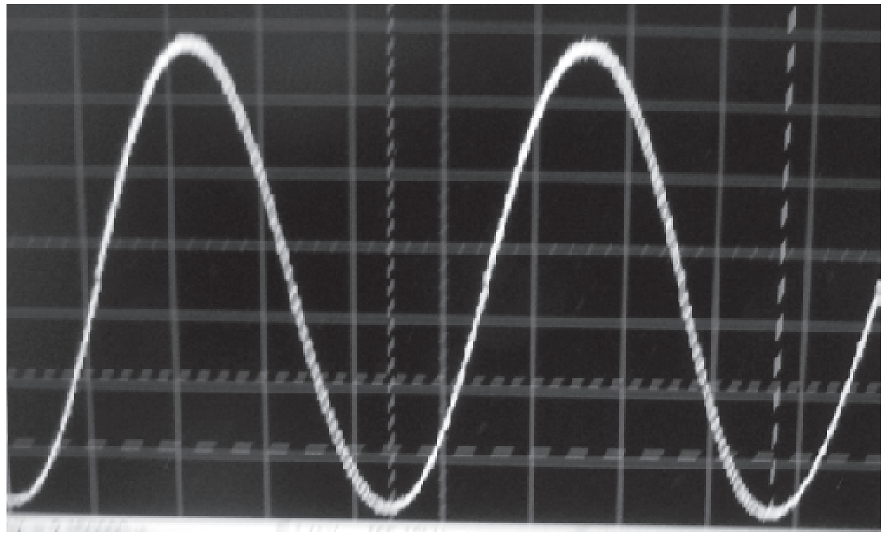


**Figure14 — This oscilloscope photo shows the output waveform with the transmitter operating at 480 kHz.**

## Results

The waveforms that the DAC produced exceeded my expectations. In the time domain, the most evident distortion that exists is caused by the sample-and-hold nature of the DAC. This distortion increases as the frequency of the output increases. In addition, a small amount of glitch energy is present, but this would not have a significant impact on the system. Figure 14 shows the output waveform at 480 kHz. Figure 15 is the output waveform with the transmitter operating at 5.8 MHz.

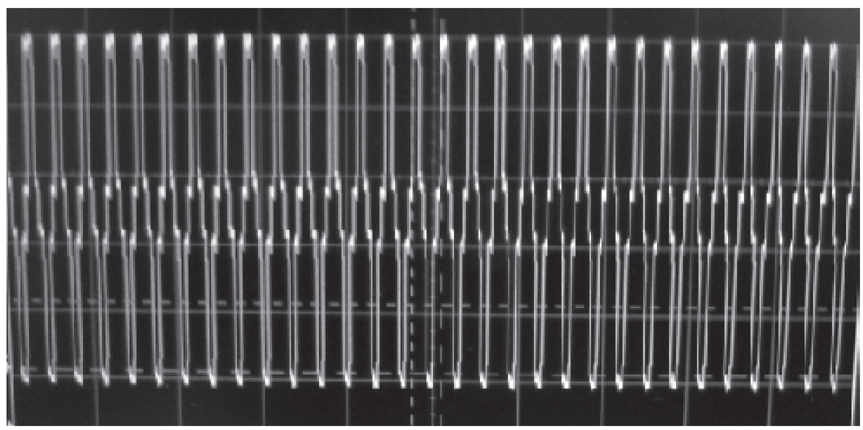


**Figure 15 — This oscilloscope photo shows the output waveform with the transmitter operating at 5.8 MHz. Note that this should correspond to the "worst case" waveform, as discussed in the text.**

Looking at the spectrum of the waveforms can give a little more insight into the performance of the radio. Bandwidth measurements may not be highly accurate because the spectrum was created by having a 200 million samples per second oscilloscope take fast Fourier transforms (FFT) of the waveform. It should be sufficient, however, to provide a good idea of the functioning of the transmitter. Figure 16 shows the 66 kHz output in both time and frequency domains.

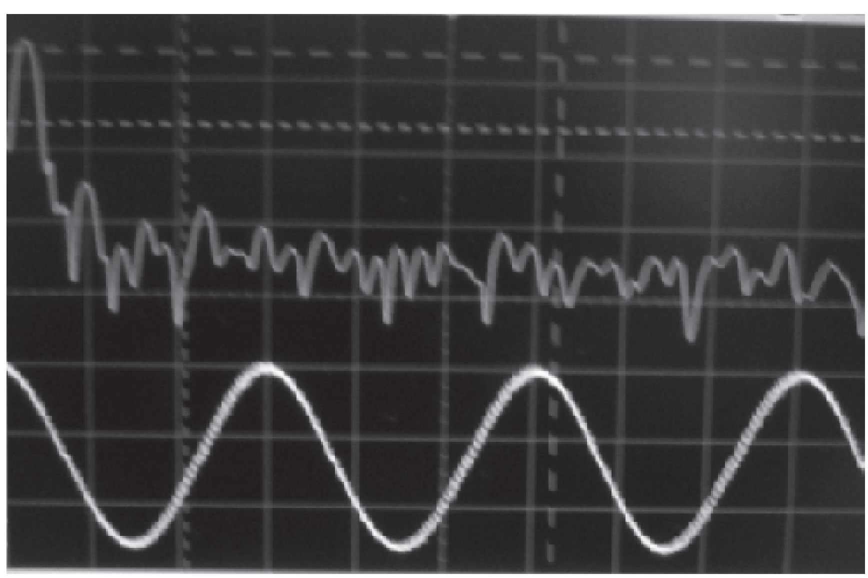


**Figure 16 — Here is an oscilloscope photo of a 66 kHz wave and its corresponding spectrum. The peak of the spectrum is over 60 dB greater than other products.**

Figure 17 shows perhaps the worst case output of the radio. A product with significant amplitude lies within the 6 MHz bandwidth of the radio. The primary product is 1.36 MHz. The first major product that is produced after that is 30 dB lower than the intended product and at about three times the original frequency. This is an alias of the original signal. Figure 18 shows the spectrum of the highest frequency generated by the transmitter, at about 6.2 MHz.

Figure 19 shows a product that occurs outside of the 6 MHz passband of the radio. This image occurs at three times the fundamental frequency and is 20 dB lower than the intended product. Since this lies outside the passband of the radio, it would be easy to remove from the final signal.

## Conclusion

This project was intended to explore how to develop an FPGA-based transmitter, as well as spread spectrum systems. The waveforms that we created show that it is very possible to use simply an FPGA and a DAC to create direct RF signals, or to serve as an IF signal in a transmitter. The waveform that was created here could easily be improved. One significant improvement would be to increase the clock speed within the system. Altera claims that the DE2 board can be driven as fast as 150 MHz. At this speed, timing would become a big issue. Increasing the clock speed would allow a much faster sample rate, which would in turn reduce aliasing and increase the maximum frequency that can be created on the board. Additionally, the size of the look-up table that was used could be significantly increased since the Cyclone II chip has just under 500 Kbits of internal SDRAM. With both of these factors in mind, it should be possible to directly synthesize signals in the entire HF band on the DE2 board. Working with the DE2 board makes experimentation very simple since no soldering is required. Creating a receiver would be slightly more difficult since the board does not contain a high-speed ADC. It is possible that one could be interfaced via the GPIO expansion port. Alternatively, a narrowband signal could be down-converted and then processed with the audio ADC on the board.

Spread spectrum systems are not commonly heard within the amateur bands. This is primarily because most of the advantages of spread spectrum come from their resistance to jamming and their stealth nature. Neither of these are particular concerns in the amateur community. FCC Rules do provide for experimentation with spread spectrum in the 70 cm and shorter wavelength amateur bands, however.

*Tom Dean, KB1JIJ, has been licensed since 2003. He currently attends the United States Military Academy at West Point where he is studying electrical engineering. Tom serves as the president of the Cadet Amateur Radio Club, W2KGY. He has taught Amateur Radio license classes and serves as the Liaison VE for exam sessions given by the club. In his free time, besides being on HF, he enjoys running, swimming and reading. His academic interests include software defined radio, satellite communications and complex variable methods in partial differential equations.*

## Notes

[1]Robert C. Dixon, *Spread Spectrum Systems,* New York: Wiley International, 1984, p 71.

[2]Ibid, p 72.

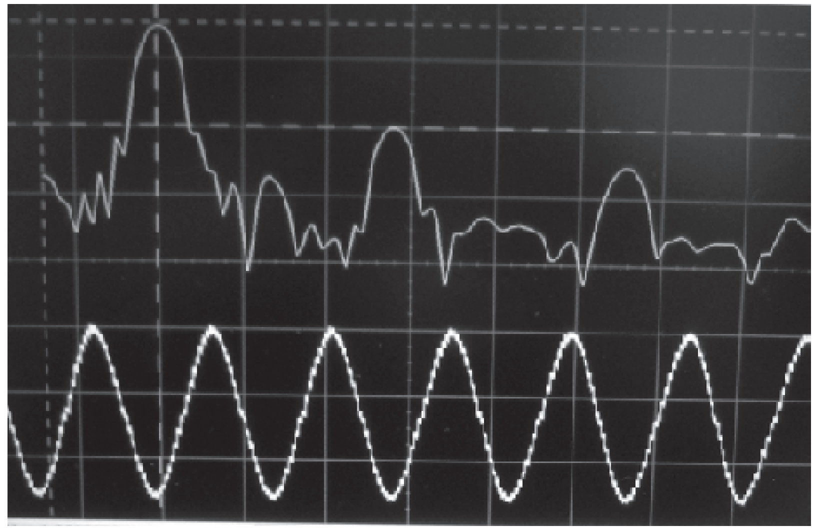


**Figure 17 — This photo is of a 1.36 MHz output. The first unwanted product is 30 dB lower than the output of the fundamental frequency.**

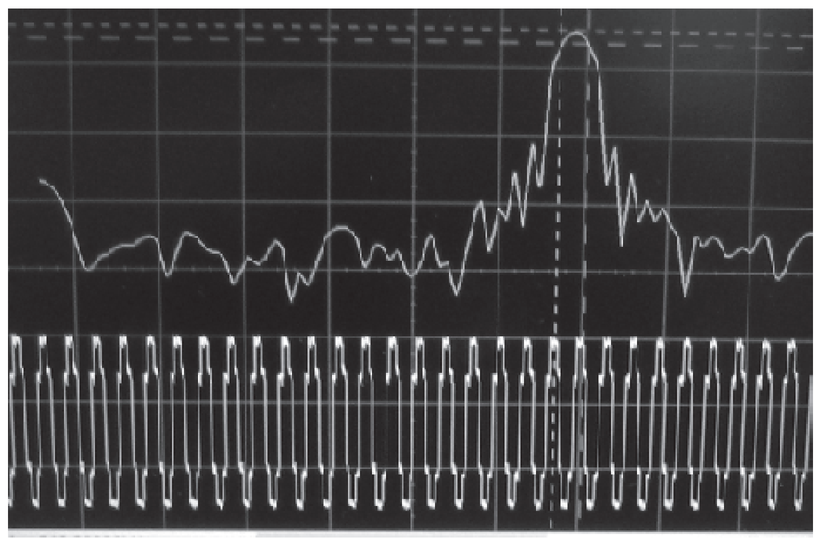


**Figure 18 — Here is the spectrum of the highest frequency generated by the transmitter. The 3 dB bandwidth of this signal is on the order of 300 kHz.**

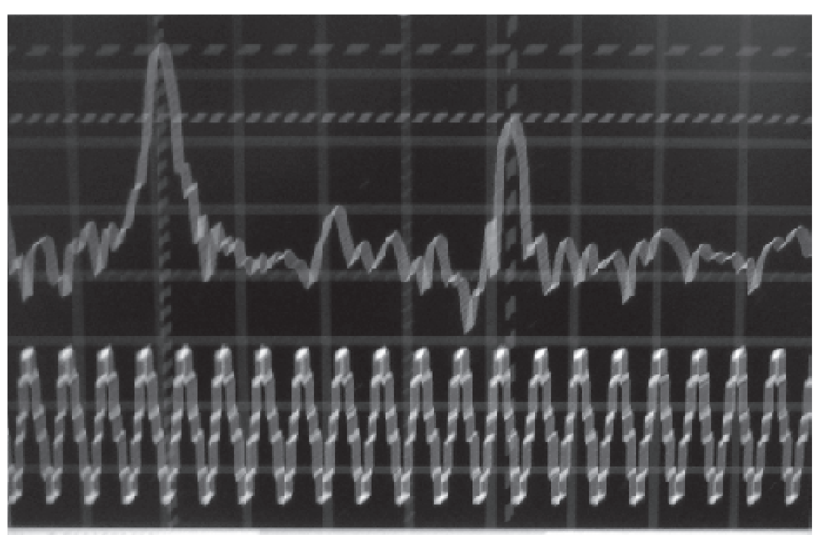


**Figure 19 — This oscilloscope display shows the transmitter operating at 4.16 MHz, with a 12 MHz image at –20 dB.**

[3]Ibid, p 58.

[4]Ibid, p 87.

[5]Anatol Tirkel, *Cross-Correlation of m-Sequences,* Australia: Monash University.

[6]Mark Wilson, K1RO, ed, *The ARRL Handbook for Radio Communications,* Newington: ARRL, 2009, p 16.2.

[7]Charan Langton, ed, *Coding and Decoding with Convolutional Codes,* Complextoreal, 1999.

[8]Ibid.

[9]Chip Flemming, *A Tutorial on Convolutional Coding with Viterbiti Decoding,* Spectrum Applications, 2006. Accessed from: **home.netcom.com/~chip.f/viterbi/tutorial.html**.

[10]Intel Corporation. *Intel ECC Application Note 108,* San Jose: Intel, 1999.

[11]George Aliftiras, *CDMA Waveform Generation for Digital Radios*, Filtronic Sigtek, p 1.

[12]*Synthesizable Sine Wave Generator,* Doulos. Accessed from: **www.doulos.com/knowhow/vhdl_designers_guide/models/sine_wave_generator/**.

[13]George Aliftiras, *CDMA Waveform Generation for Digital Radios*, Filtronic Sigtek, p 4.

[14]Ken Yang, *Flatten DAC frequency response,* Maxim Integrated Products, 2006. Accessed from: **www.edn.com/article/CA6321533.html**.

[15]George Aliftiras, *CDMA Waveform Generation for Digital Radios*, Filtronic Sigtek, p 4.

**Rudy Severns, N6LF**

PO Box 589, Cottage Grove, OR 97424; n6lf@arrl.net

# Experimental Determination of Ground System Performance for HF Verticals Part 7 Ground Systems With Missing Sectors

*Here is the author's research on radial systems that do not make a full circle around the vertical antenna.*

A very common problem with vertical ground systems is the impracticality — in many situations — of laying down a symmetric circle of radials. Some object, frequently a structure or a property line, may make it impossible to place radials in certain areas around or near the base of the antenna. I have received many questions on this subject so I decided to do some experiments where I compared the signal strength ($S_{21}$) of a ¼ λ vertical antenna that has a full 360° radial fan to one with a substantial portion of the radial fan missing in one sector.

The first part of the experiment was done at four frequencies: 7.2, 14.2, 21.2 and 28.5 MHz. The second part the experiment was done at 7.2 MHz only.

## Radial Fan Configurations

For this series of tests I chose to use a symmetric 360° radial fan with thirty two 33 foot radials (¼ λ on 40 m) as the reference configuration (C1). As shown earlier in this series, a radial system with thirty two ¼ λ radials is usually pretty good. You can add more radials, but the gain is relatively small, so a 32-radial system is a good compromise, and probably more typical of amateur installations. The radials were close to ¼ λ on 40 m. Figure 1 shows a plan view of the initial radial fan geometries.

The four 180° sectors were arranged in relation to the receiving antenna as follows:

1) Radials toward (C2),
2) Radials away (C3),
3) Radials to the left (C4), and
4) Radials to the right (C5).

Both right and left configurations, which ideally should be identical, were run as a check on the consistency of the measurements.

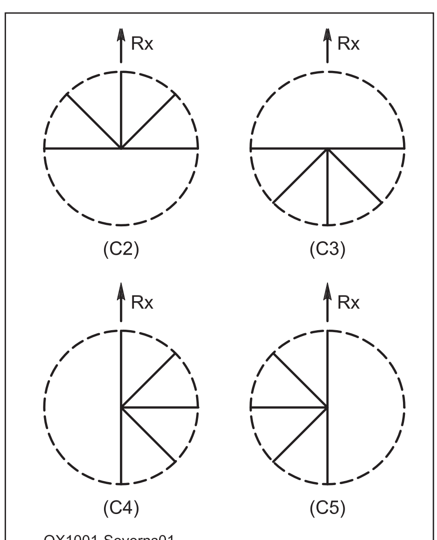



**Figure 1 — Missing sector radial layouts.**

After running tests using configurations C1 through C5, I realized that some additional radial configurations might be interesting. In particular I wanted to see how much adding some short radials in the missing sector would improve things.

I added the configurations shown in Figure 2 to the experiment:

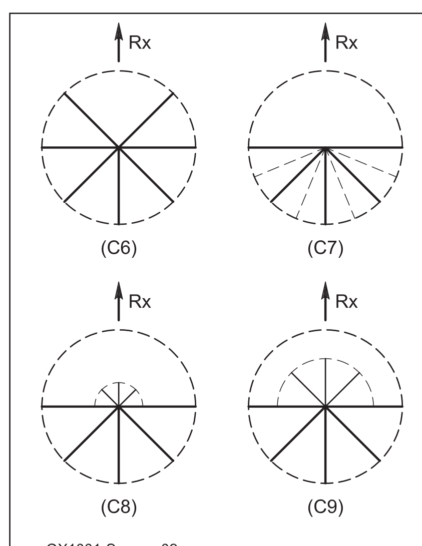



**Figure 2 —Additional asymmetric ground systems.**

**Table 1**

**Effect of a 180° Sector Ground System on Signal Strength ($S_{21}$) in a Given Direction Relative to the Receive Antenna**

| *Frequency (MHz)* | *C2 Toward RX (dB)* | *C3 Away from RX (dB)* | *C4 Left (dB)* | *C5 Right (dB)* |
|---|---|---|---|---|
| 7.2 | –0.42 | –1.91 | –0.82 | –0.94 |
| 14.2 | –0.57 | –2.42 | –1.20 | –1.24 |
| 21.2 | –0.69 | –3.00 | –1.24 | –1.33 |
| 28.5 | –0.55 | –3.23 | –1.26 | –1.58 |

**Table 2**
**$S_{21}$ Test Results for the Added Radial Configurations**

| *Radial Configurations* | *$\vert S_{21}\vert$ Referenced to C1 (0.0 dB)* |
|---|---|
| C6 | –0.44 |
| C3 | –1.91 |
| C7 | –1.39 |
| C8 | –1.52 |
| C9 | –0.34 |

5) A 90° missing sector (7 radials removed, 25 radials remaining) (C6). The axis of the missing sector was pointed at the receiving antenna.

6) To C3, which has 17 radials facing away, I added an additional sixteen 33 foot radials between the seventeen already there (33 radials total) (C7). The missing 180° sector was facing the receiver.

7) To C3 I added fifteen 8.5 foot radials in a fan towards the receiving antenna. These are 1⁄16 λ radials on 40 m (C8). C9) To C3 I added fifteen 17 foot radials in a fan towards the receiving antenna. These are 1⁄8 λ radials on 40 m.

## Test Results

Modeling ground systems with missing sectors using *NEC* indicates that compared to a full 360° system we should see both a reduction in the peak signal and a distortion in the pattern; in other words, a front-to-back ratio not equal to 0 dB.

Experimental results are given in Tables 1 and 2. Note that Tables 1 and 2 show the *difference* in dB from the 360° radial fan (C1), which is the reference.

Clearly sector radial systems have an impact on the radiated signal. In the direction of the remaining radials the signal loss is on the order of 0.5 dB, but in the direction of the missing sector the loss is from 1.9 to over 3 dB. If you have a 3 dB loss, that means you have lost half your power. Not good!

The test results qualitatively agree with *NEC*, the peak amplitude is reduced and the pattern is distorted when only a partial radial fan is employed. The radial system used for the tests reported in Table 1 has 33 foot radials, which of course are long for frequencies above 7.2 MHz. As we saw in the discussion for multi-ground systems (Part 6), the system with all 40 m radials gives the best performance, even better than if we used thirty two ¼ λ radials tailored for each band.

The test results for radial configurations C6 through C9 are given in Table 2. All of these tests were done at 7.2 MHz.

The first thing we see is that omitting the seven radials in a 90° sector (C6) does not seem to do too much harm, only –0.44 dB. Eliminating all the radials in a 180° sector (C3) is not good, however (–1.91 dB). The loss jumps by almost 1.5 dB over the 90° case!

Taking the radials removed from C1 (to form C3) and adding them between the remaining radials in C3 (C7) helps a little bit, reducing the loss by 0.5 dB. If, instead, we add fifteen 1/16 λ radials (C8) in the missing sector we get a similar improvement, about 0.4 dB. Despite some improvement, the signal loss for both C7 and C8 is still substantial. What really seems to help is to put fifteen 1/8 λ radials (C9) in the missing sector. Unfortunately, that may not always be possible.

## Some Closing Comments

Overall, it's pretty clear both from modeling and experiment that sector ground systems can reduce your signal substantially in some directions and produce a distorted pattern.

What can we do about this? The first thing is to remember that the field intensity around the vertical increases rapidly as we get near the base of the antenna.[1] If we move the base of the antenna away from the obstacle as little as 1/16 λ or better yet 1/8 λ, so that we can have at least some radials in the sector towards the obstacle, the losses will be reduced. As shown above, 1/8 λ spacing can be quite effective. In the process of moving the base away from the obstacle you may have to shorten some of the other radials on the side away from the structure but that may be acceptable. Another possibility would be to move the base from the side of the building to a corner which might allow the radial fan to be increased from 180 to 270°. As the test data shows, this can be very helpful.

These experiments were done in an ideal situation. There was no actual structure next to the antenna. In addition to the losses we see in this idealized situation, it is very likely that the structure blocking the radial fan will increase the loss. It is difficult to estimate how much the loss will increase, but it's not likely that the building will improve your signal! Another factor to consider is the soil characteristics. My soil, over which these tests were conducted, would be rated as good or even very good, depending on the time of year. Poorer soils would result in even larger negative effects due to the use of a sector ground system than those shown in Tables 1 and 2.

What I have shown here represents only a few of many possibilities. It's not possible to experimentally examine all possible situations, but *NEC* modeling should give you a good qualitative feeling for your particular situation. One common situation that I did not have time to examine experimentally is the case where the base is alongside the house but not too far from a corner. The conventional wisdom is that you should run the radials along the side of the house to the corner and then fan them out from there. I don't think that can hurt but keep in mind that the farther you are from the corner, the less effective this scheme is likely to be.

[1]Rudy Severns, N6LF, "Verticals, Ground Systems and Some History," *QST*, Jul 2000, pp 38-44.

*Rudy Severns, N6LF, was first licensed as WN7WAG in 1954 and has held an Extra class license since 1959. He is a consultant in the design of power electronics, magnetic components and power-conversion equipment. Rudy holds a BSE degree from the University of California at Los Angeles. He is the author of three books and over 90 technical papers. Rudy is an ARRL Life Member, and also an IEEE Fellow.*

**Paul Wade, W1GHZ**

161 Center Rd, Shirley, MA 01464; w1ghz@arrl.net

# Waveguide Filters You Can Build—*and* Tune Part 2 Waveguide Post Filters

*In this second of his three part series, the author introduces us to a handy technique for waveguide filter design and construction.*

In all but the simplest microwave systems, filtering is needed, to eliminate both undesired radiation and unwanted interference. For most amateurs, this means copying some published filter. Simple filters often have inadequate performance, while more complex ones can be difficult — in understanding, fabrication or tuning. Simple waveguide filters can be designed to provide excellent performance, using available software and fabricated in a modestly equipped amateur workshop.

Waveguides have very low loss because the energy is contained inside the guide, in air, rather than traveling in a conductor. A resonant length of waveguide, with very low loss, thus forms a high-*Q* resonator; for X-band waveguide, the theoretical *Q* approaches 10,000.[1] This high unloaded *Q* enables the design of very sharp filters with low loss. Since only metal and air are involved, and the waveguide dimensions are tightly controlled, results are quite predictable.

If you need a review of filter terminology and design basics, please see Part 1 of this series, "Filter Tour," in the Nov/Dec 2009 issue of *QEX* for a brief filter overview with minimal mathematics.

The waveguide filters to be described here are direct-coupled resonator filters.[2] The *WGFIL* program by Dennis Sweeney, WA4LPR, does an excellent job of designing either *iris* or *post* filters.[3] An iris filter is more intuitive — each waveguide iris is a perpendicular wall across the guide, so that two irises create a resonant cavity in a section of waveguide. A small hole in each iris provides coupling out of the cavity, with coupling controlled by the size of the hole. Fabrication involves cutting very thin slots across the waveguide and soldering an iris in each slot.

Waveguide post filters are much easier to

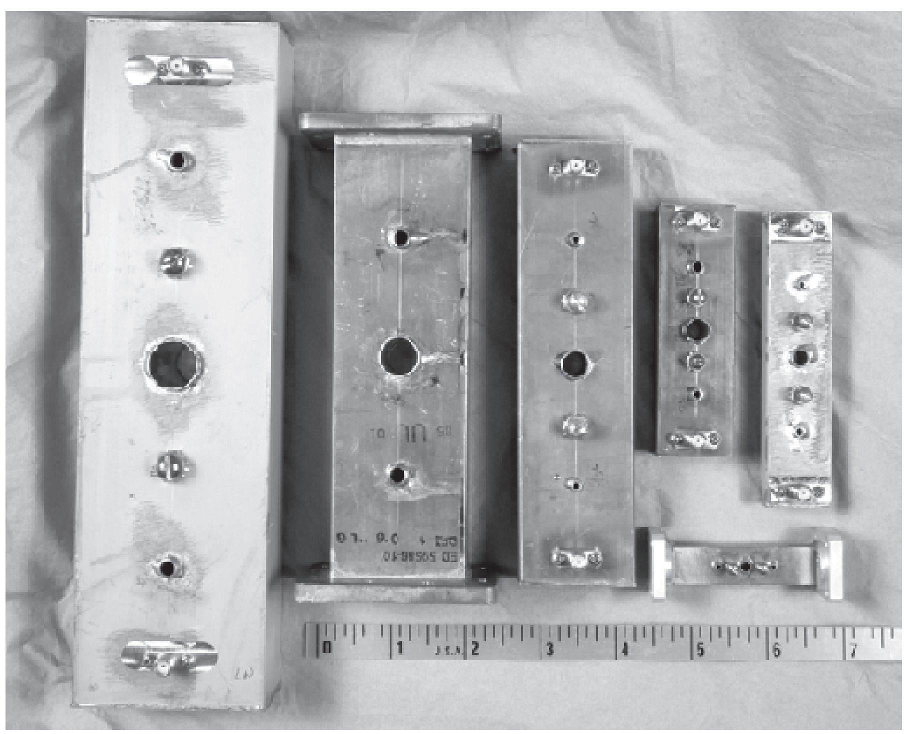


**Figure 1 — This photo shows waveguide post filters in many sizes of waveguide.**

[1]Notes appear on page 25.

build — just drill a hole through the center of the wide dimension, insert a post all the way through, and solder both ends. Design is a bit more complicated: a post is a shunt inductance in the waveguide, which acts both as a cavity wall and as an impedance inverter, coupling the adjacent cavities. A larger post blocks more of the waveguide, so coupling is reduced by larger posts. The most difficult calculation is the distance between posts for a desired resonant frequency, since the diameter of the post also affects the resonance. The *WGFIL* program does an excellent job here, and I have made several filters that perform exactly as expected; a few of them are shown in Figure 1. All my successful filters are tabulated in Table 1 for those who prefer to duplicate a proven design.

Since a waveguide filter is not hard to build, and results are predictable, there is a temptation to design a really high performance filter, with multiple sections. The filter may be easy to build, but it is really difficult to tune — we must allow for some tuning to compensate for construction tolerances. A multi-section filter has extremely good stop-band rejection, and a mistuned filter has *no* passband — if nothing detectable gets through, then it is nearly impossible to do any tuning. I have a nice six-section filter that I built for 24 GHz that looks great on the computer, but I've never been able to tune it properly, even with a fancy Vector Network Analyzer (VNA).

Some very good waveguide post filter designs have been published — I can recommend filters by N6GN for 10 GHz and 5.76 GHz.[4, 5] These are three section (four-post) filters with good performance that can be built and tuned by a reasonably well-equipped microwaver. Figure 2 shows the performance of two 10 GHz filters built and tuned in my basement — all the data shown is measured with a nice Rohde & Schwarz *ZVA* VNA (**www.rohde-schwarz.com**) set up by Greg Bonaguide, WA1VUG, at the Eastern VHF/UHF Conference in 2008.

The easy-to-build waveguide post filters use surplus waveguide and posts of brass or copper hobby tubing, available at some hardware stores and hobby shops, or online from **www.smallparts.com**. The hobby tubing comes in increments of 1⁄32 inch diameters, so only a few of the smallest sizes are suitable for the higher microwave bands, particularly 10 GHz and 24 GHz. Using stock sizes severely limits filter design to a very few bandwidths, particularly for multiple-section filters that require several different post diameters.

## Simple Double-Tuned Filters

While I was playing with *WGFIL* trying to find a better filter using stock tubing diam-

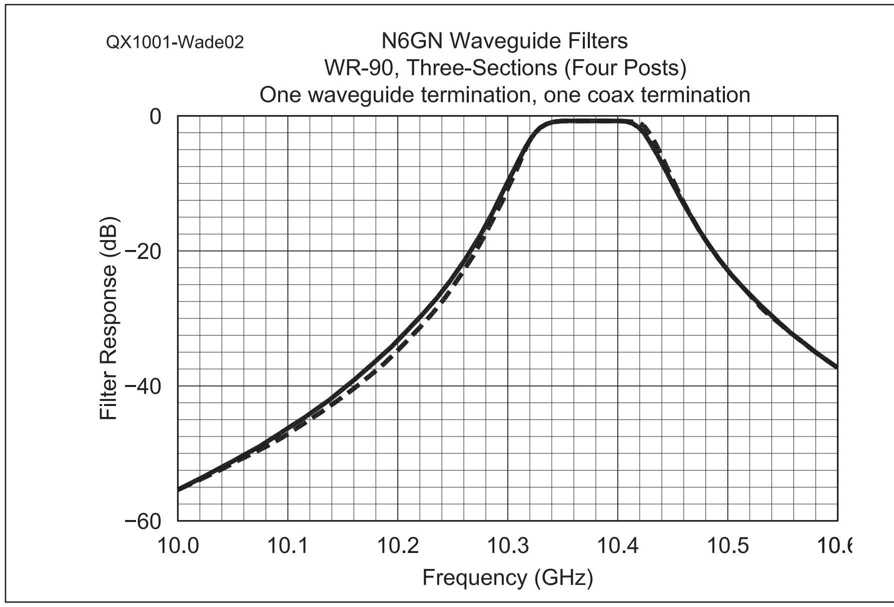



**Figure 2 — A graph of the performance of N6GN 10 GHz waveguide filters.**

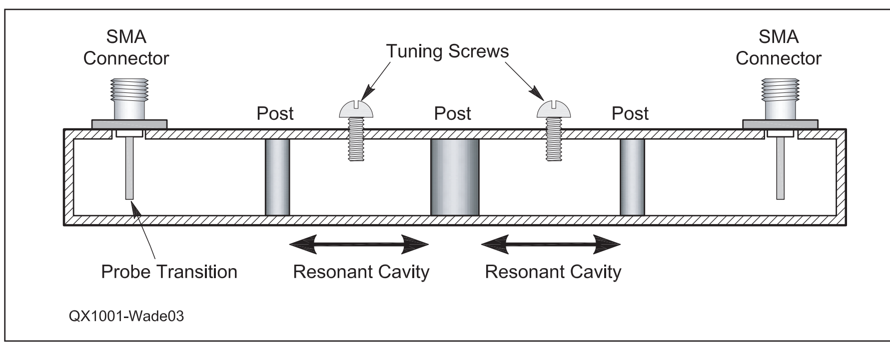



**Figure 3 — This sketch represents a three-post (two-section) waveguide filter.**

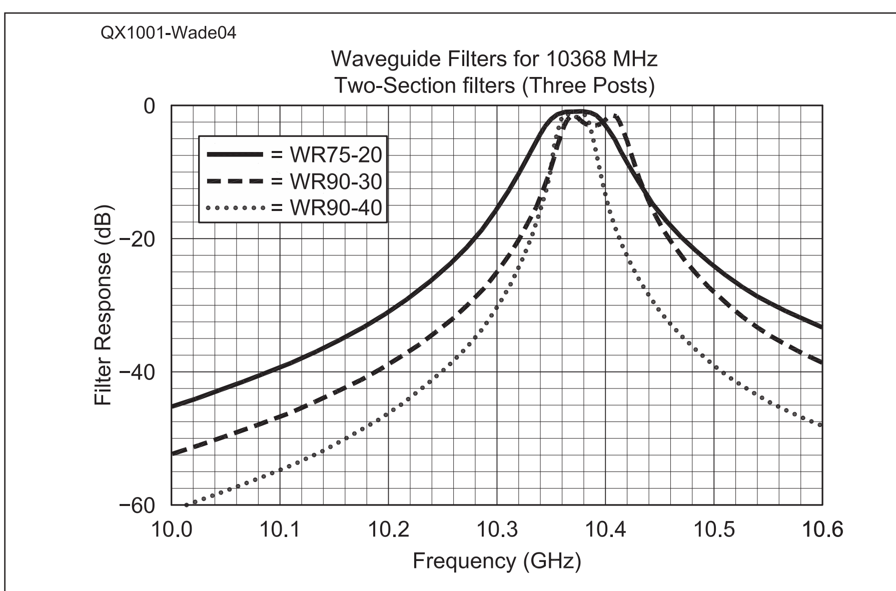



**Figure 4 — The performance of two-section (three-post) waveguide filters for 10 GHz are shown in this graph.**

eters, it occurred to me that hams don't need multiple-section filters with steep skirts. Our main requirement is to reject LO leakage and mixer images. The most popular intermediate frequency is 144 MHz, so a usable filter needs perhaps 30 dB of rejection at a frequency 144 MHz away from the operating frequency, like the ones shown in Figure 2. A balanced mixer provides some additional rejection, and the image frequency, twice as far removed, will be further down.

This requirement doesn't seem too difficult. Perhaps something as simple as a double-tuned filter would be adequate. It could be a narrow, high-*Q* filter, because waveguide has low loss and a double-tuned filter has only two tuning adjustments, and they should be identical. Some trial runs with *WGFIL* for a two-section (three post) filter suggested that several possibilities existed using the available hobby brass. Figure 3 is a sketch showing the simplicity of these filters. This sketch includes integral waveguide-to-coax transitions.

I did a bit of simulation using the Ansoft *HFSS* electromagnetic software, then built a couple of filters for 5760 and 10,368 MHz.[6] These worked so well that I tried some other waveguide sizes and frequencies. Measured results are shown in Figure 4 for 10,368 MHz, with filters in both WR-90 and WR-75 waveguide, with different bandwidths. All provide adequate LO rejection for a 144 MHz IF at 10,368 MHz, with about 0.7 dB of loss at the operating frequency. The narrower two, WR75-20, with 20 MHz bandwidth, and WR90-30, with 30 MHz bandwidth, appear over-coupled. The latter one is tuned to one of the peaks, the one providing best rejection, rather than centered. We shall see later how to adjust a filter design for coupling that yields a flatter response.

Figure 5 plots three filters for 5760 MHz, in two different waveguides, WR-137 and WR-159, with different bandwidths. All of these provide at least 35 dB of LO rejection for a 144 MHz IF, at 5760 MHz, with about 0.5 dB of loss at the operating frequency. The version in WR-159 waveguide is slightly overcoupled, and tuned to one of the peaks, rather than centered.

Waveguide filters for 3456 MHz are relatively large, but provide excellent performance, as shown in Figure 6. Loss of these filters is less than ¼ dB, with a flat response, and LO rejection for a 144 MHz IF is more than 30 dB.

All of these two-section filters were easily tuned using basement test equipment, since there are only two tuning screws, and the screws should have identical settings. Also, the dimensions were chosen for ease of tuning, so that tuning would only require a small penetration by the tuning screws.

## Over-Coupled Filters

The response of most of these filters has a nice flat top, with reasonably steep skirt — an ideal Maximally-flat, or Butterworth,

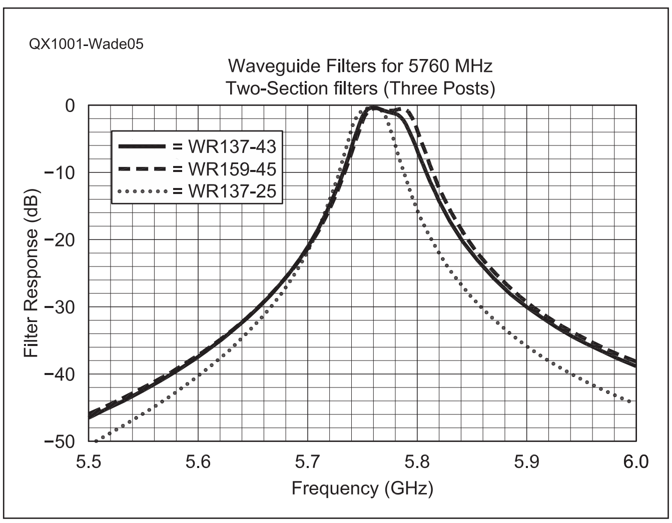



**Figure 5 — This graph shows the performance of two-section (three-post) waveguide filters for 5760 MHz.**

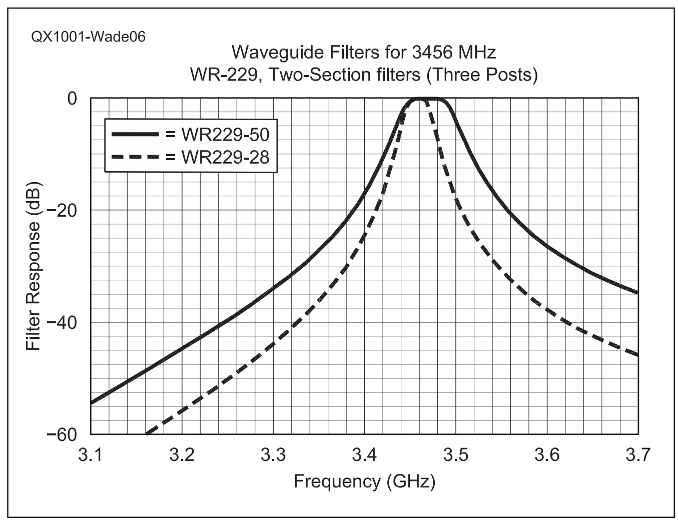



**Figure 6 — The performance of two-section (three-post) waveguide filters for 3456 MHz are shown in this graph.**

**Table 1**
**Three-Post Waveguide Filters W1GHZ 2008**

| *Waveguide* | *Frequency (MHz)* | *Bandwidth (MHz)* | *End Post Diam (Inches)* | *Mid Post Diam (Inches)* | *Spacing (Inches)* | *Data* | *F - 144 (dB)* |
|---|---|---|---|---|---|---|---|
| WR-75 | 10368 | 20 | 0.125 | 0.250 | 0.950 | WR75-3-20 | –42 |
| WR-90 | 10368 | 30 | 0.188 | 0.313 | 0.860 | WR90-3-30 | –34 |
| WR-90 | 10368 | 40 | 0.156 | 0.313 | 0.830 | WR90-3-40 | –28 |
| WR-137 | 5760 | 25 | 0.188 | 0.406 | 1.620 | WR137-3-25 | –41 |
| WR-137 | 5760 | 43 | 0.156 | 0.375 | 1.600 | WR137-3-43 | –34 |
| WR-159 | 5760 | 45 | 0.250 | 0.500 | 1.480 | WR159-3-45 | –33 |
| WR-187 | 3456 | 7 | 0.250 | 0.438 | 3.000 | (not built) | –50 |
| WR-229 | 3456 | 50 | 0.188 | 0.500 | 2.500 | WR229-3-50 | –33 |
| WR-229 | 3456 | 28 | 0.250 | 0.625 | 2.540 | WR229-3-28 | –41 |
| WR-42 | 24192 | 140 | 0.094 | 0.156 | 0.360 | WR42-3-140 | –15 |
| *With Coupling Screw* | | | | | | | |
| WR-75 | 10368 | 20 | 0.125 | 0.281 | 0.970 | WR75-3c-20 | –45 |
| WR-75 | 10368 | 42 | 0.094 | 0.250 | 0.940 | WR75-3c-42 | |
| WR-42c | 24192 | 70 | 0.094 | 0.188 | 0.375 | WR42-3c-70c | –20 |

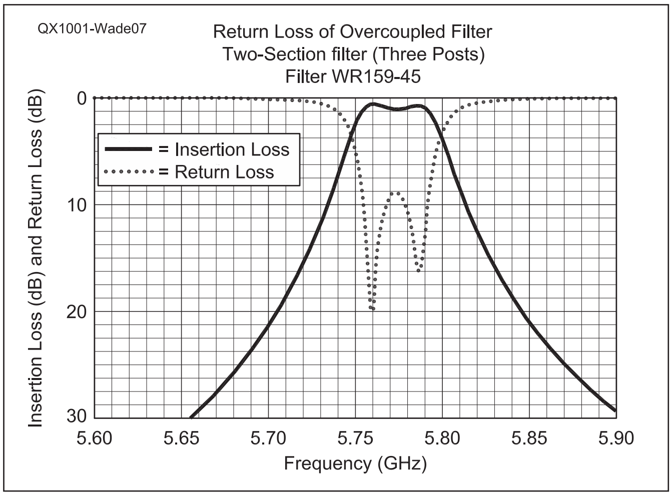



**Figure 7 — This graph illustrates the double-humped response of an overcoupled waveguide filter.**

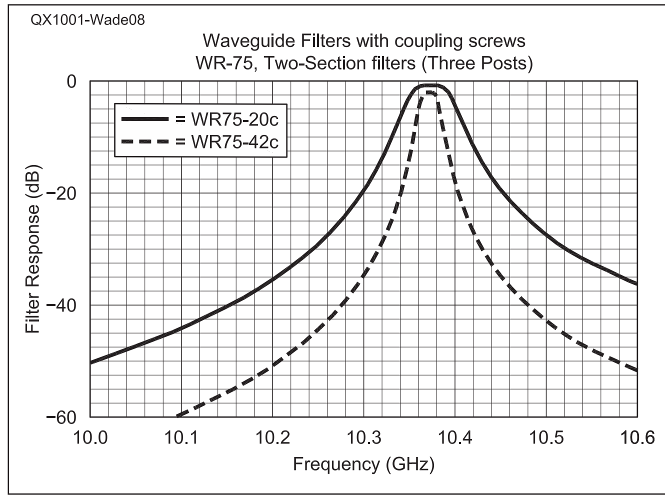



**Figure 8 — The performance of waveguide post filters with coupling screws, adjusted for a flat response are shown in this graph.**

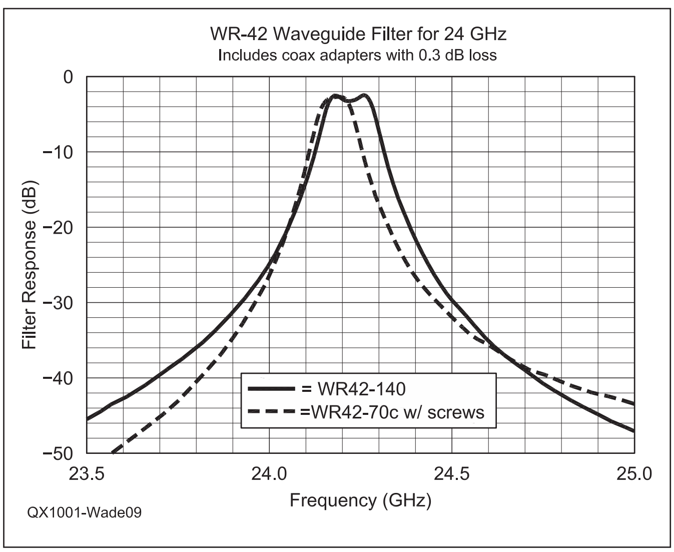



**Figure 9 — This graph shows the performance of two-section (three-post) waveguide filters for 24 GHz, with and without a coupling screw.**

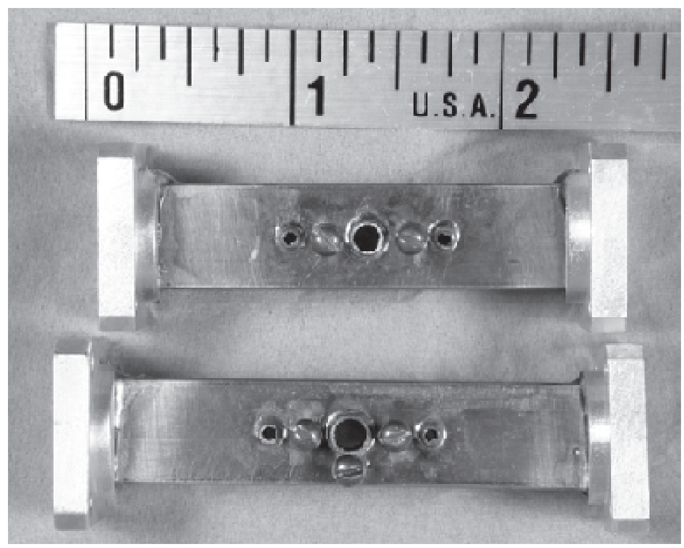



**Figure 10 — This photo shows three-post waveguide filters for 24 GHz, in WR-42 waveguide.**

filter. A couple of them show an overcoupled response, with a dip in the middle, however. These are a little harder to tune, since it requires picking the hump that gives the best LO rejection, and adjusting accordingly. The return loss is particularly sensitive to overcoupling; in Figure 7, the return loss is very good at the hump frequencies, but not as good in between. With ideal coupling, the return loss would be good over the whole passband. The *WGFIL* calculations are pretty good, but not perfect, especially for larger post diameters; that's why it gives a warning when the posts are bigger than ¼ of the waveguide width. As a result, we don't always get perfect coupling, especially when we round off the diameter to the nearest 1⁄32 of an inch.

I really wanted to make a reproducible 24 GHz filter, since I don't know of any that have been published. Trial runs with *WGFIL* weren't as promising, since only two or three sizes of hobby tubing are small enough, and I only found one promising combination. I considered using AWG wire sizes — copper wire is readily available — but the diameters weren't right. Some commercial filters use multiple small posts rather than a large one, but a bit of research didn't find any simple answers for designing them.

I recalled that some commercial filters have an extra screw next to each post, and I wondered if that varies the coupling. I went back to *HFSS* to find out. What I found was that the extra screw increases the coupling, in effect making the post *smaller*. This was the answer! I also found that the coupling screw must be inserted a long way, nearly half the waveguide height, to have a significant effect, so that small adjustments should be easy. I designed and built two more 10 GHz filters in WR-75 waveguide, with slightly oversize center posts, to try out the coupling screw. The coupling screw should decrease the effective size of the center post to adjust the coupling to the desired response. It worked perfectly, as shown in Figure 8. Both filters are adjusted for a flat response and centered on the operating frequency. The wider one, with 42 MHz bandwidth, has lower loss, about 0.75 dB. The narrow one, with 20 MHz bandwidth, has higher loss (about 2 dB) but is sharp enough to provide about 20 dB of LO rejection for a 30 MHz IF at 10,368 MHz. It is not surprising that the sharper filter has more loss, since a higher loaded *Q* is needed for the narrower bandwidth.

The coupling screw is next to the center post, halfway between the post and the side wall of the waveguide. While it would be possible to put a coupling screw next to the other posts and make them adjustable also, tuning would no longer be straightforward. For the simple three-post filters, it is unnecessary, and each screw adds a small additional loss.

Then I made two filters for 24 GHz, in WR-42 waveguide. One was the best combination of available post diameters I could find, while the other has an oversize center post and a coupling screw. The results are shown in Figure 9. Both filters are sharp enough to use for a 144 MHz IF at 24 GHz. The lower one, with the coupling screw, has a nice flat response; the other is slightly over-coupled and has a bandwidth slightly wider than expected as a result. Each has about 2.5 dB of loss. That's not bad for a sharp filter at 24 GHz. These filters were tuned up at a single frequency, 24.192 GHz, since I don't have a sweeper for 24 GHz. The plotted data was later measured with the VNA without any retuning. Figure 10 is a photo of the 24 GHz filters.

With the addition of the coupling screw, tuning these filters becomes very easy. Starting with all screws all the way out, the two tuning screws are slowly inserted simultaneously (turn one, then the other the same amount) until some output is found. Then peak the output. Since the response is under-coupled without the coupling screw inserted, there will only be a single peak. Next, insert the center coupling screw; the output will slowly increase, then start to decrease as the response becomes over-coupled with a dip in the middle. Backing the screw out to the peak yields the desired flat response. A final trim probably won't make much difference.

The tuning progression is illustrated in Figure 11, a simulation of the WR75-42c filter. The curve on the right shows the response before tuning, with no screws present. The filter is tuned to some higher frequency. The tuning screws alone move the response down to the desired frequency, yielding the curve labeled "No Coupling Screw." Then the coupling screw is inserted; at 0.100 inch deep, the response flattens and the loss is reduced. Inserting the screw farther produces an over-coupled response, first with a slight dip at the operating frequency, and then a huge dip if insertion is continued. Most of us would back up when the output started to dip.

Some of the filters in Figures 4 and 5, as well as one of the 24 GHz filters, show an over-coupled response. These designs could be improved by making the center post one size (1⁄32 inch) larger and adding a coupling screw next to the center post. Then they could be adjusted for a flat response.

## End Termination

In a waveguide system, these filters only need waveguide flanges to connect. Most systems for 10 GHz and lower frequencies use semi-rigid coax for interconnections, however, so a coax-to-waveguide transition is needed. The most compact and convenient transitions are integral to the filters, one at each end. I use the transition dimensions that I published in *QEX*, spacing the transition probe at least one waveguide width from the end post.[7] A matching screw is neither needed nor desired. If the dimensions are correct, the return loss will be very good. Of course, a badly mismatched component following a filter can upset the filter response, but the place to correct this is not in the filter.

The filters with performance plotted in Figure 2 provide a good comparison. One has integral coax transitions, while the other has waveguide flanges and was tested with external transitions. Any slight difference in performance is probably due to construction tolerances and tuning difference.

## Construction

These filters are physically simple to build. The posts, tuning screws and coax connectors are all on the centerline of the broad dimension of the waveguide. Important points are that the posts be accurately centered on the centerline and that the holes for the posts are snug, so that a minimal amount of solder is needed to make a good connection.

The highest frequency for each resonator is set by the distance between the posts. A tuning screw can only lower the frequency. The distances calculated by *WGFIL* are with no tuning screw, so they should be reduced slightly to raise the resonant frequency and allow a small amount of tuning. I estimate that I can locate a hole within 10 mils (0.25 mm), so I reduce the distance by 10 to 15 mils. Adjust the dimensions for your filters according to the tolerances you can achieve. With only a small reduction, a very few turns penetration of the tuning screw is needed. A larger reduction in spacing will require more penetration, increasing losses and making the tuning more critical.

I measure and mark the centerline and hole positions with a cheap caliper, either dial or digital, using the points as a scribe (this would be criminal abuse with a quality tool). Then the holes are marked with a center punch and started with a small center drill. A drill press is essential for drilling the holes. For accurate, round holes that fit the posts snugly, I find that DeWalt "Pilot Point" drill bits work well; Black & Decker "Bullet" drills are nearly as good. For larger holes, Unibit step drills work very well. A small pilot hole drilled through both sides of the waveguide will allow making the larger holes

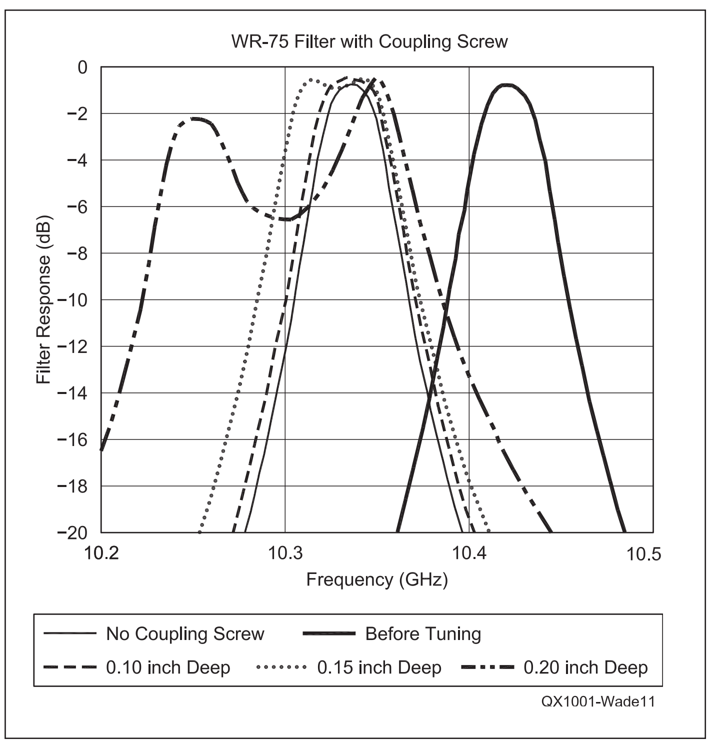



**Figure 11 — This graph illustrates the tuning progression for a waveguide filter with a tuning screw.**

from opposite sides.

Screw holes are tapped using the drill press to keep them square, turning by hand with the motor unplugged. Then the burrs inside the waveguide are cleaned up using a fine file.

The outside of the guide and the posts is cleaned using a Scotchbrite pad — the coarser brown variety may be needed for old, badly oxidized, waveguide.

If the posts fit snugly, they will need to have one end chamfered slightly so that they may be pressed in – if they are loose enough to fall out, the filter will probably still work but it may have higher loss. Then resin paste flux is applied around the ends of each post where they project from the waveguide. Finally, a single ring of thin solder is wrapped around each end of the posts and pressed into the flux to hold it in place. Tin-lead eutectic (63-37) solder is preferred — it will flow smoothly and at a slightly lower temperature than other solders.

## Ends

If the filter includes an integral waveguide-to-coax transition, the ends must be closed with a short circuit. I use a plate of hobby brass a bit larger than the waveguide outside dimensions, so that the plate has a bit of overhang. I paint the ends of the guide, which have been filed square, with solder flux, then put the end plates on and clamp them in place. Scraps of firebrick or ceramic tile insulate the clamps from the end plates. Finally, I wind a ring of solder around the waveguide and press it into the flux to hold it in place.

Preparation for waveguide flanges is similar.

## Soldering

For soldering with soft solder, I prefer a hot air gun to a torch. A hot air gun, the kind used for stripping paint, has no flame and doesn't get as hot, so the metal oxidizes less.

I've had good results preheating the filter assembly on a hot plate to near soldering temperature, and then applying the hot air gun to each area being soldered. A few seconds after the hot air is applied to a spot, the ring of solder around the joint will melt and flow into the joint. As soon as the solder flows around the whole ring, move on to the next joint. When all the joints have been flowed, use gloves or hot pads to gently move the assembly onto a firebrick or other heat-tolerant surface to cool slowly.

## Summary

The filters described here are intended to provide good performance with minimum complexity, so that they are easy to design and to tune. These waveguide filters offer high performance but do require some metalworking. Some proven designs are tabulated and the *WGFIL* software is sufficient to design custom filters.

All the filters described here are designed for "good enough" performance at a particular microwave ham band. Good enough means that commonly used LO frequencies and mixer image frequencies are suppressed by at least 20 dB, and more than 30 dB in most cases. This should be adequate to radiate a clean signal and to suppress out-of-band interference.

Part 3 of this series will present "Evanescent Mode Waveguide Filters."

### Notes


[1]G. F. Craven and C. K. Mok, "The Design of Evanescent Mode Waveguide Bandpass Filters for a Prescribed Insertion Loss Characteristic," *IEEE Transactions on Microwave Theory and Techniques*, March 1971, pp 295-308.

[2]Ralph Levy, "Theory of Direct-Coupled-Cavity Filters," *IEEE Transactions on Microwave Theory and Techniques*, June 1967, pp 340-348.

[3]Dennis G. Sweeney, WA4LPR, "Design and Construction of Waveguide Bandpass Filters," *Proceedings of Microwave Update '89*, ARRL, 1989, pp 124-132. The *WGFIL* program may be downloaded from **www.w1ghz.org/filter/WGFIL.COM**

[4]Glenn Elmore, N6GN, "A Simple and Effective Filter for the 10-GHz Band," *QEX*, July 1987, pp 3-5.

[5]Paul Wade, N1BWT, "A Dual Mixer for 5760 MHz with Filter and Amplifier," *QEX*, Aug 1995, pp 9-13. A PDF file of this article is available at **www.w1ghz.org/10g/QEX_articles.htm**.

[6]See the Ansoft Web site at **www.ansoft.com**

[7]Paul Wade, W1GHZ, "Rectangular Waveguide to Coax Transition Design," *QEX*, Nov/Dec 2006, pp 10-17. A PDF file of this article is available at **www.w1ghz.org/10g/QEX_articles.htm**.


QEX

## Next Issue in *QEX*

ARRL President Joel Harrison, W5ZN, and Robert McGwier, N4HY, describe "The Design, Construction and Evaluation of the 8 Circle Array for Low Band." This 160 meter array was first designed by Tom Rauch, W8JI, and is described in the fourth edition of *ON4UN's Low Band DXing* by John Devoldere. Joel and Bob present a step by step guide to building, tuning and using this antenna array. This steerable receiving antenna system requires some significant real estate, but offers excellent performance for the effort to build it.

**Maynard Wright, W6PAP**
6930 Enright Dr, Citrus Heights, CA 95621; w6pap@arrl.net

# *Octave* for Bessel Functions

*The author continues to share new ways we can use this versatile program for our electronics calculations*

With the advent of digital voice modes for repeaters, there has been renewed interest in the bandwidth of FM signals, as methods for getting more out of the repeater allocations are explored. At its July 2009 Board of Directors Meeting, ARRL authorized a committee to study narrowband channel spacing for the VHF/UHF bands.[1]

In light of this newly relevant emphasis on FM bandwidth, we might be interested in the bandwidth of the transmitters we own, whether to just think about the matter in a more informed way or in order to be able to measure what's really going on.

Bessel functions of the first kind give us a convenient way to measure or calculate the bandwidth of an FM signal.[2] A chart of Bessel functions appears in Figure 8.9 on page 8.6 of the 2010 *ARRL Handbook*.[3] That graph is reproduced here as Figure 1. If we use a sine wave to modulate an FM transmitter and adjust the modulation so that the carrier or one of the sidebands goes to zero amplitude, Figure 1 will give us the modulation index for that particular input level to the transmitter. Some useful modulation indexes are tabulated in the Figure caption.

Didn't we do something similar to this in *Octave for Signal Analysis*?[4] In that article we used the Fast Fourier Transform (FFT) of *GNU Octave* to determine the magnitudes (and phases) of the various spectral components of an FM signal, and then we graphed them.[5]

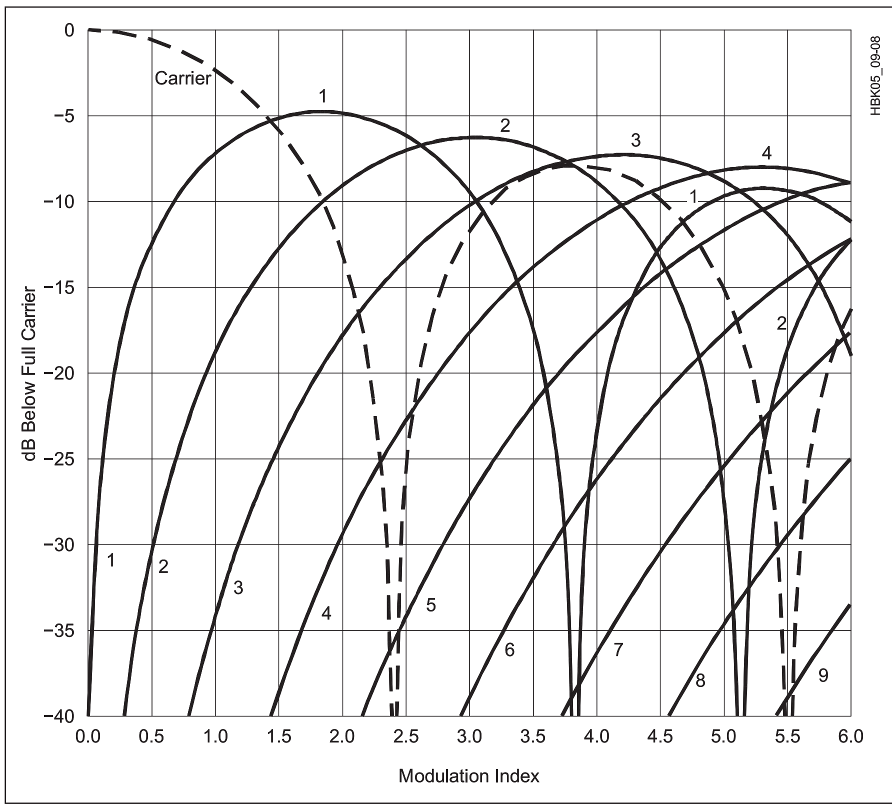



**Figure 1 — This graph illustrates how the amplitude of the FM carrier and the first nine sidebands vary with modulation index. The values are plotted in terms of decibels below the unmodulated carrier. This is a graphical representation of the mathematical functions developed by F. W. Bessel. Note that the carrier completely disappears at modulation indices of 2.405 and 5.52.**

Why two seemingly different mathematical techniques that yield (almost) the same results? Fourier analysis and Bessel function series expansions are different ways to express the same information. Like computer languages, or written language, sometimes one is better than the others for the current activity. They both have important applications in heat transfer, spectral analysis, filter design, and other areas of science and engineering. Both are examples of generalized Fourier series. Their usefulness overlaps in the area of FM signal analysis.

For examining the entire spectrum of an FM signal, Fourier analysis is the right tool. Fourier analysis can also handle modulating waveshapes other than sinusoids. For seeing what happens to the sidebands as the modulation index changes, if we can restrict our analysis to a sinusoidal modulating signal, a graphical presentation of Bessel functions such as Figure 1 is more useful and intuitive.

If, however, we want to precisely determine the amplitudes of the various sidebands, the *GNU Octave* FFT will give us more precision than will Figure 1. As an alternative to using an FFT, though, we

[1]Notes appear on page 29.

might write some *Octave* code to calculate the Bessel functions displayed in that graph, and to determine any of the zero crossings of any of the sidebands with considerable precision and accuracy.

A problem with the Bessel function, though, is that it involves solving a differential equation that doesn't have an exact, or closed form, solution so we must take another approach. (See Note 2.) We'll use an infinite series to represent the Bessel functions we want. Once we've done that for Bessel functions, we'll have added a powerful tool to our software toolbox that can be used to approximate the values of many other useful functions.

It's better to call our method a truncated infinite series because we can't carry out the addition of the terms infinitely and, as is the case for all convergent infinite series the terms become smaller and smaller as we progress, eventually becoming small enough to ignore for any particular purpose.[6] We'll include a test to stop the calculation and addition of terms after we achieve the precision we need.

The series that we'll use is:[7]

$$y = \sum_{m=0}^{\infty} \frac{(-1)^m \, t^{(v+2m)}}{2^{(v+2m)} m! \, \Gamma(v+m+1)}$$

(Eq 1)

In our *Octave* code, we'll calculate the value following the summation sign over and over again, adding the new incremental value to the old sum until the difference between two successive values of y becomes small enough so that we exit the loop. We can ease the load on the computer by moving values out of the loop that are always the same. If we factor Equation 1 to do that, we get:

$$y = \frac{t^v}{2^v} \sum_{m=0}^{\infty} \frac{(-1)^m \, t^{(2m)}}{2^{(2m)} m! \, \Gamma(v+m+1)}$$

(Eq 2)

If we want to run this code under a math utility other than *Octave* that doesn't feature the Gamma function, we can make use of the fact that $\Gamma\,(v + m + 1)$ equals $(v + m)!$ if we restrict $v$ to integer values. (See Note 7, p 270.)

The code to calculate Bessel functions is listed in Table 1. We'll use *while* and *endwhile* to begin and end the loop. The test in the *while* statement prevents us from locking up the computer by looping endlessly. We'll calculate the results of the summation first and then multiply by the term we factored out. The parentheses around the –1 that is exponentiated are important. Without the parentheses, the normal operator precedence of *Octave* will cause $1^m$ to be calculated first, before the application of the negative sign, and that's not what we want. We want "minus one raised to the *m*th power," not "the negative of the quantity 'one raised to the *m*th power'." The correct version here reverses the sign with every succeeding term.

Readers interested in learning more about *Octave* may wish to check out the *Octave* mailing list. Send an e-mail to **mailman-request@octave.org** with just the word help in the message body. You will receive a reply with instructions on how to proceed.

Now that we can calculate FM sideband amplitudes to any precision we need, how about the zero crossings? These are very useful and two of them are given to four significant figures in the caption of Figure 1. We'll use a technique called the Newton-Raphson Method to calculate the precise location of any zero crossing of any of the sidebands. (See Note 6.) We'll find that this technique is easy to apply and is useful in many other applications where we need to find the argument of a function that produces a particular value of that function.

We'll ask the user for the sideband number and an initial estimate from the keyboard. We'll then use *Octave*'s function *besselj*, rather than our own function from Table 1, to simplify the code.

What? There's a function available to calculate the Bessel function in *Octave*? Well, yes: we wrote our own code to show ourselves how to use the truncated infinite series and to give us some confidence in *Octave*'s internal function. If we add a line: printf("\n BESSEL FUNCTION AMPLITUDE FROM besselj() = %g", besselj(v, t)); to the code in Table 1, we can compare our routine with *Octave*'s and we'll see that they yield the same results.

Another reason for developing our own routine for Bessel functions is that they are pretty useful, not only for frequency modulation calculations, but for filter design and analysis. Some of the other math utilities we studied in *Alternatives to Octave* are not equipped with an internal Bessel function routine.[8] We can now move to any one of them with confidence that we can employ Bessel functions as we like, and that we can write series approximations for other missing functions if we need to do so.

In our quest for zero crossings, we'll refer to Figure 2. Figure 2 doesn't look much like Figure 1. It is, in fact, an approximation to

**Table 1**
**Bessel Function of the First Kind**

```
printf("\n           BESSEL FUNCTION CALCULATIONS");
v = input("\n\n            SIDEBAND NUMBER: ");
t = input("           MODULATION INDEX: ");

Bessel = 0; temp = 1; m = 0;
while (abs(Bessel - temp) > 1e-7)
  temp = Bessel;
  Bessel += ((-1) ^ m * t ^ (2 * m)) / (2 ^ (2 * m) * factorial(m) * gamma(v + m + 1));
  m = m + 1
endwhile
Bessel *= (t ^ v / 2 ^ v);
Bessel_dB = 20 * log10(abs(Bessel));
printf("\n         BESSEL FUNCTION AMPLITUDE = %g", Bessel);
printf("\n      BESSEL FUNCTION AMPLITUDE IN dB = %g\n\n", Bessel_dB);
```

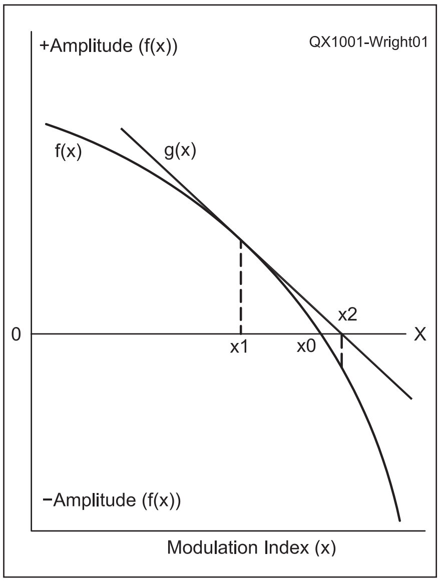



**Figure 2 — This graph illustrates the Newton-Raphson Method of determining the modulation index of an FM signal.**

**Table 2**
**Newton-Raphson Method for Finding Zero Crossings**

```
printf("\n          ZERO CROSSING CALCULATIONS");
sideband = input("\n\n     SIDEBAND NUMBER = ");
x = input("INITIAL ESTIMATE OF ZERO = ");
y = besselj(sideband, x);
while (abs(y) > 0.0001)
  y = besselj(sideband, x);
  dy = (besselj(sideband, x + 0.000001) - y) / 0.000001;
  x = x - y / dy;
endwhile
printf("\n  ZERO CROSSING (x) = %g", x);
printf("\n      ERROR (y) = %g\n\n", y);
```

one of the curves from an earlier *Handbook* in which the levels were not converted to dB but were expressed as amplitudes relative to the unmodulated carrier.[9] The code in Table 1 prints out values that correspond to the "raw" amplitude ratios of the earlier figure as well as to the values in dB below the unmodulated carrier of the more recent figure. The Newton-Raphson method won't work well with the discontinuous curves of the newer figure, although that figure is more convenient when we want to eyeball what's happening to the sideband levels.

We'll first estimate a value *x1* for the zero crossing, which actually occurs at *x0* in Figure 2. We'll then either calculate or approximate the value of the first derivative, *m*, of the function at *x1*. Since analytic differentiation of the Bessel function is difficult at best, we'll approximate the derivative by calculating values of the amplitude at values of *x* that are very close to each other and by letting the first derivative be represented by:

$$f'(x1) = (f(1) - f(a)) / (x1 - xa) \qquad \text{[Eq 3]}$$

where *xa* is a point along the *x* axis very close to *x1* and *f(n)* is the value of the function at *x = n*.

We then construct a line, *g(x)*, that is tangent to the curve at *f(x1)* that passes through *x = 0*. The equation for such a line is:

$$g(x) - g(x1) = f'(x1) \times (x - x1) \qquad \text{[Eq 4]}$$

The line represented by that equation is *g(x)* in Figure 2. If we set *g(x)* equal to zero we can solve for the associated value of *x* and we can see from Figure 2 that it will be closer to the actual value *x0* than is *x1*. We'll then do it over again and achieve an *x* that is even closer to *x0*. After a few repetitions, we'll have a value of *x* that will be as close to *x0* as we like. As our estimates move closer and closer to the actual value, they may or may not cross to the "other side" of the correct value along the *x* axis, depending on the direction of concavity of the curve and where our original estimate fell.

The code to implement this procedure is listed in Table 2. (The *Octave* code files from this article are available for download from the ARRL *QEX* website.)[10] The code loops using the above procedure while moving closer and closer to the desired value. We'll use a *while* loop to stop the procedure when we are "close enough." What is close enough? For this calculation, we'll assume that when the value of the amplitude (represented by *y* in the code) is less than 0.0001, we ought to stop and be happy with the result. The command *while (abs(y) > 0.0001)* tests *y* every time the *endwhile* command returns execution to the top of the loop. The program exits the loop and continues on through the remaining code when the test condition is no longer true. We'll print out *x* to give us our zero crossing and *y* to show us the error in attempting to reach *y* = 0.

We can change the precision to something other than 0.0001 if we like. We can also use the Newton-Raphson method in other applications where we want to find zero crossings of functions or, by adding an extra constant to the equation, to find at what argument a function passes through a particular value. As we've seen before, it often requires very little code to implement a procedure that would be pretty complicated and tedious if we were to do it manually.

As we've discussed before, we need to note carefully that the code here will work fine for personal use where we know what we're doing. We've left out some safeguards that are considered good programming practice because we don't want such details to obscure what we're doing here and we would probably more than double the size of the code if we included everything we ought to. A couple of relatively standard precautions would be:

1. Test inputs to make sure that they are valid. When we enter the modulation index, for instance, the program ought to issue a diagnostic message and reject the input if we input a letter or a string;

2. Provide some way to exit the *while* loop, in case something goes wrong, to prevent locking up the computer. A simple counter that exits after, say, one hundred iterations would be a good safety feature. The following pseudocode *while* loop is protected in that way:

```
counter = 0;
while(something is true)
  do some work;
   counter++; # increment the counter
  if(counter > 99)
   break;
endwhile
```

counter++ increments the counter and when the work specified in the loop has been done one hundred times, execution will jump from the loop to the statement following the endwhile statement.

A problem with the Newton-Raphson method that affects its use with curves such as Bessel functions is that the initial guess value for *x* needs to be selected relatively close to the zero crossing we'd like to find. There are some formal requirements for the function being evaluated that take care of that (see Note 6), but we'll discuss the main difficulty here to see how it affects our use of this technique. If we select a value of *x* near a loop (maximum or minimum value) of one of the curves, the first derivative will yield a relatively flat line that may find an initial zero crossing far beyond several other crossings that are not close to the one we are seeking. The succeeding iterations will home in on a zero crossing, but it probably won't be the one we want.

If, for example, we evaluate sideband number 1 and select an initial estimate of 2.0, we get a value $x$ = 10.1735. This is a valid zero crossing, but it's not very close to the two zero crossings that bracket our initial guess.

We'll try out our code by locating the third zero crossing of the carrier with an initial estimate of 8.5, obtained from eyeballing Figure 1. We'll specify sideband 0 to get the carrier. The results are $x$ = 8.65373 and $y$ = $4.69302 \times 10^{-5}$. This value of $x$ corresponds nicely to the third zero crossing, 8.654, as shown in the caption of the *1986 ARRL Handbook*, Chapter 9, Figure 3. (See Note 9.) That graph is reproduced here as Figure 3.

We'll also try out the use of the code to find a level other than a zero crossing. Let's say that we would like to know the modulation index for which the carrier level is dropped 6 dB. That corresponds (approximately) to a voltage ratio of 0.5, so we'd like to find the modulation index for which the carrier drops to 0.5 relative to its unmodulated amplitude. From inspection of Figure 1, we estimate that the crossing of that amplitude will occur at a modulation index of about 1.5. We can use Newton-Raphson by bringing the point on the curves that normally corresponds to 0.5 down to 0. We do that by subtracting 0.5 from the returns of all of the calls to function *besselj* in the code. When we do that we get a crossing of 0.5 amplitude at a modulation index of 1.52114 with an error in $y$ of $-3.07862 \times 10^{-5}$. If we choose to do so, we can generalize the code to permanently include the $y$ offset and to accept a user input of the offset, which would be zero if we want to find a zero crossing.

The Newton-Raphson method gives us a powerful tool for finding arguments corresponding to specified outputs for a wide variety of functions from various applications.

*Maynard Wright, W6PAP, was first licensed in 1957 as WN6PAP. He holds an FCC General Radiotelephone Operator's License with Ship Radar Endorsement, is a Registered Professional Electrical Engineer in California, is an ARRL Member, and is a Life Senior Member of IEEE. Maynard has been involved in the telecommunications industry for over 45 years. He has served as technical editor of several telecommunications standards and holds several patents. He is a Past Chairman of the Sacramento Section of IEEE. Maynard is Secretary/Treasurer and past President of the North Hills Radio Club in Sacramento, California.*

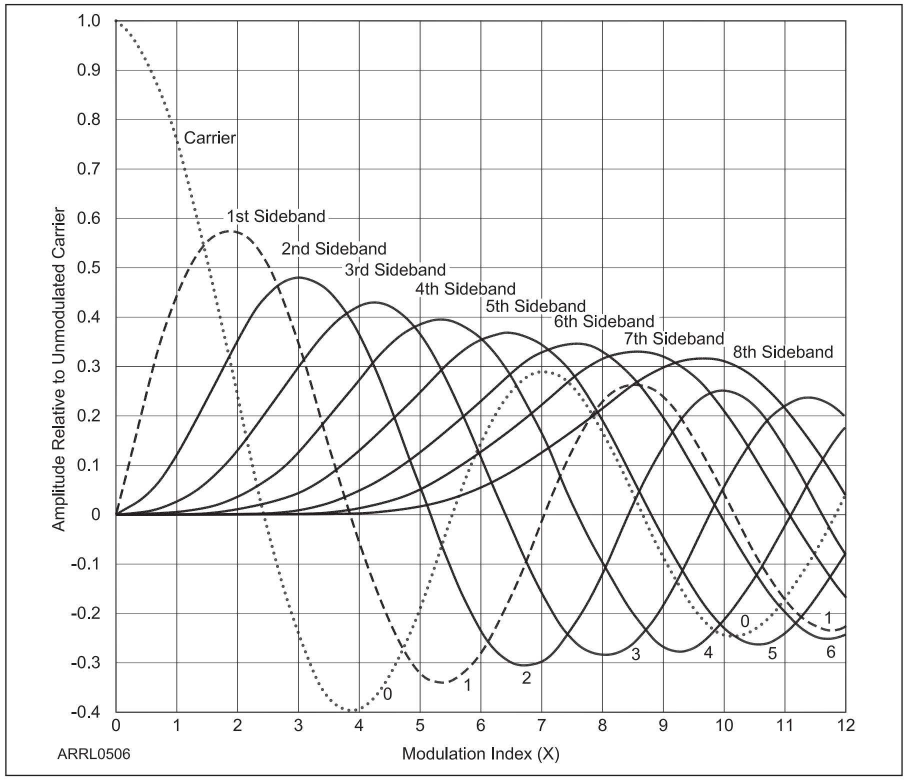



**Figure 3 — This graph illustrates how the amplitude of the FM carrier and the first eight sideband pairs vary with modulation index. The values are plotted in terms of amplitude relative to the unmodulated carrier. This is a graphical representation of the mathematical functions developed by F. W. Bessel. Note that the carrier completely disappears at modulation indices of 2.405, 5.52 and 8.654.**

## Notes:


[1]S. Khrystyne Keane, K1SFA, Ed, *The ARRL Letter*, Vol 28, No 29 (Friday, July 24, 2009), available at **www.arrl.org/arrlletter/09/0724/**.

[2]B. P. Lathi, *Modern Digital and Analog Communication Systems,* Third Edition, Oxford University Press, 1998, page 222.

[3]Mark Wilson, K1RO, Ed., *The ARRL Handbook for Radio Communications*, 2010 Edition, ARRL, 2009, Figure 8.9. p 8.6. *The ARRL Handbook* is available from your local ARRL dealer, or from the ARRL Bookstore, ARRL order no. 1448. Telephone toll-free in the US 888-277-5289, or call 860-594-0355, fax 860-594-0303; **www.arrl.org/shop**; **pubsales@arrl.org**.

[4]Maynard Wright, W6PAP, "*Octave* for Signal Analysis*", QEX*, Jul/Aug, 2005, pp 25-

[5]See **www.octave.org**.

[6]W. L. Hart, *Analytic Geometry and Calculus*, D C Heath and Company, 1957, pages 294-296.

[7]C. Ray Wylie, *Advanced Engineering Mathematics*, Fourth Edition, McGraw-Hill, 1975, page 397.

[8]Maynard Wright, W6PAP, "Alternatives to *Octave*," *QEX*, Jul/Aug, 2009, pp 25-27.

[9]Mark J. Wilson, AA2Z, Ed, *The ARRL 1986 Handbook for the Radio Amateur*, The American Radio Relay League, Inc, 1985, Figure 3, p 9-4.

[10]The *Octave* code files from this article are available for download at **www.arrl.org/qexfiles**. Look for the file **1x10_Wright.zip**.


QEX

**Ray Mack, W5IFS**

17060 Conway Springs Ct, Austin, TX 78717; w5ifs@arrl.net

# SDR: Simplified

### Reader Feedback

I got a note from Peter Anderson, KC1HR, who wrote a *QEX* article in 1994 about using the Harris Digital Down Converter ICs for an SSB receiver.[1] These ICs implement the sample rate reduction using decimation and filtering that we looked at last issue. His observation is that the ICs from Harris and Texas Instruments are fairly old and do not have enough internal precision to handle large dynamic range for narrow band applications. Many modern implementations (including the TAPR software defined radio) use an FPGA with more precision to implement the dedicated functions for sample rate reduction. The AD9864 is much newer and uses 24 bit precision internally. This part also starts its work from a much lower starting frequency than the Harris parts. We will look at using this IC in a future column.

I have heard from some of you and have started an e-mail update list. I am also planning to set up a Web site, so we have a place to post updates, corrections, and software resources between issues of the magazine. I will continue to provide files to post on the ARRL *QEX* files Web site.

### Nyquist Meets Real World

The Nyquist theorem says that you can exactly recreate an analog signal that has been sampled by passing the signal back into a digital to analog converter (DAC) as long as you select the sample rate greater than twice the highest frequency component of the input signal. The assumption is that the input analog to digital converter (ADC) and the output DAC perform identically to the sample theorem that is central to the mathematics. The real world is never so easy. In general, the ADC comes pretty close to converting the analog signal into a sequence of samples that matches how an actual impulse sample sequence would work. Of course, there are noise sources that add noise to the system, but those are just a normal part of electronic systems. We'll look at noise sources in ADC operation in the future.

The DAC system, on the other hand, suffers from a serious disconnect between the math and the electronics. The math assumes that we are going to convert a sequence of infinite height pulses with an area equal to the sample value into a

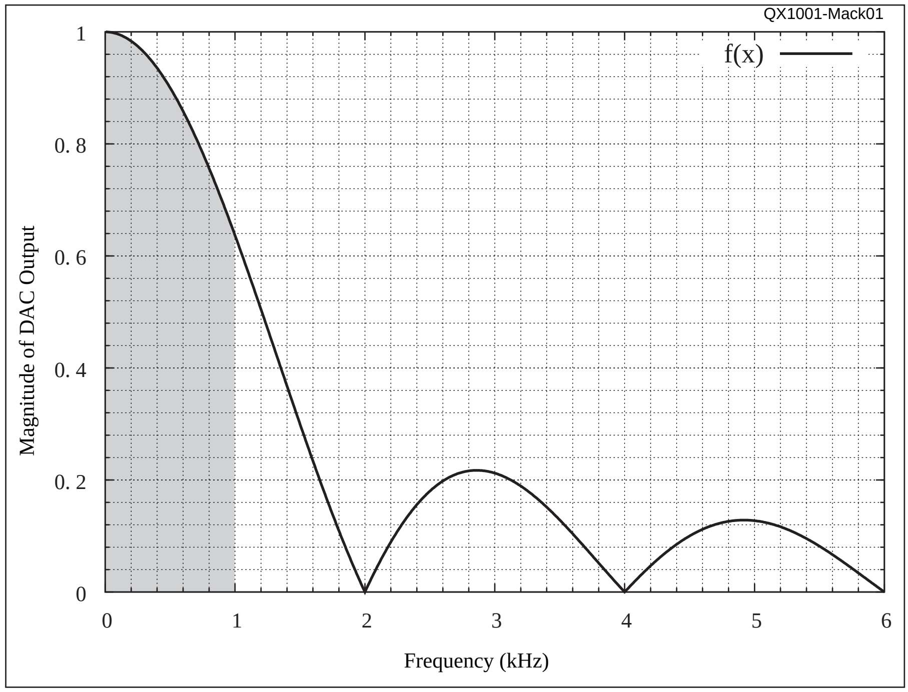



**Figure 1 — This is a graph of the frequency spectrum for 2 kHz sample rate and frequencies from dc to 1 kHz. The shaded area shows the RMS output that will be achieved at each frequency in the first Nyquist zone. The envelope is the magnitude of the sinc function.**

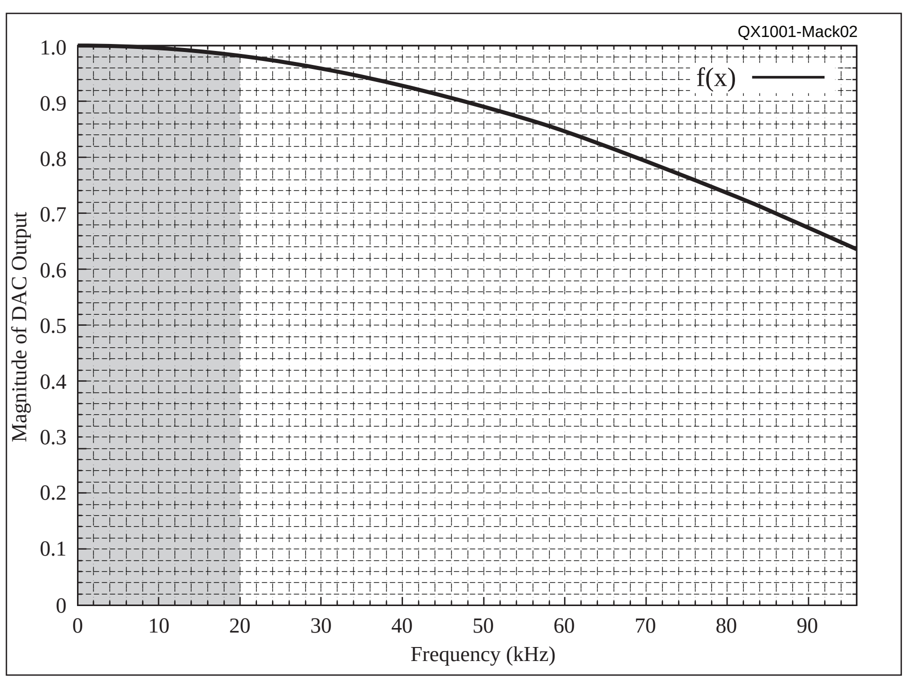



**Figure 2 — Here is a graph of the frequency spectrum for CD audio that is sampled at 192 kHz. The shaded area shows how the sinc function affects frequency. Without compensation, the frequencies near 20 kHz are only attenuated by 2% (–0.18 dB).**

[1]Peter Traneus Anderson, KC1HR, "A Simple SSB Receiver Using a Digital Down Converter," *QEX*, Mar 1994, pp 17-23.

continuous signal. A low pass filter (especially a brick wall filter) will convert those impulses into the appropriate sine waves with the proper amplitude and phase. As the low pass filter becomes less than a brick wall, the conversion allows more of the energy at higher frequencies to leak into the output and distort the waveform. So our first problem on output conversion is the fidelity of the low pass reconstruction filter.

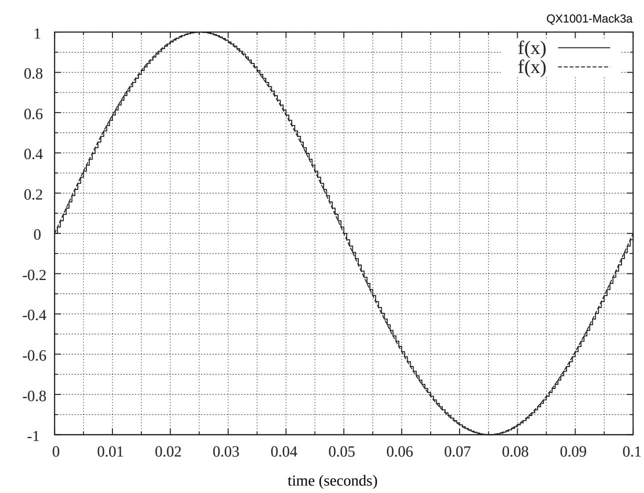
The next problem comes from the way a DAC works. Mathematicians call this a *zero order hold* circuit. You can think of the zero order hold function as an electronic transform that converts the rectangular impulse (infinite height, zero width, but finite area) into a rectangular pulse that is one sample period wide with the same area as the input impulse. Remember that the impulse sequence created a comb in the frequency domain, with all of the elements having the same height and containing all harmonics of the sample frequency, out to infinity. When you run the impulse sequence through a zero order hold (the DAC), you change the frequency content of the resulting signal. The signal still extends to infinity, but the phase and amplitude vary with frequency. That variation is defined by the function (sin x)/x. This function is called the ***sinc*** function. Enter ***wiki dac sinc function*** in your search engine to see a mathematical explanation.

Let's look at an example in which we have our DSP use a sample rate of 2 kHz, so our Nyquist area extends from dc (zero hertz) to just below 1 kHz. We will assume our system has a brick wall filter at 1 kHz. We can have our DSP create a direct digital synthesis sine wave with 1 V RMS, and vary the frequency from 1 Hz up to 999 Hz. Figure 1 shows the actual RMS output voltage we measure versus frequency. Figure 1 also shows several Nyquist zones beyond the first, so you can see the shape of the sinc function over a larger frequency range. Notice that the output voltage from the DAC and low pass filter is only 0.637 V for the 999 Hz waveform.

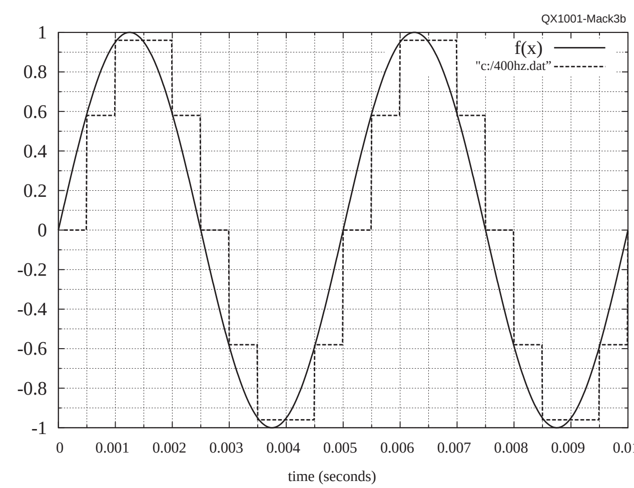
One way to solve the problem so that all frequencies from dc to 999 Hz have 1 V RMS is to change the low pass reconstruction filter to have a shape that is the inverse of the sinc function. This method is frequently used but hard to implement with analog filters. The preferred option is to apply the inverse sinc compensation to the digital samples.

The sinc function issue was a problem for CD audio (and computer audio since it originally used CD technology) when it was first introduced in the mid 1980s. The CD sample rate of 44.1 kHz was about the fastest that could easily be implemented at 16 bit resolution 25 years ago. This put the Nyquist frequency at 22.05 kHz, and allowed reasonable analog reconstruction filters that were able to cut off at the upper audio range of 20 kHz. CD audio players had to implement filters to compensate the higher frequencies because of the sinc problem.

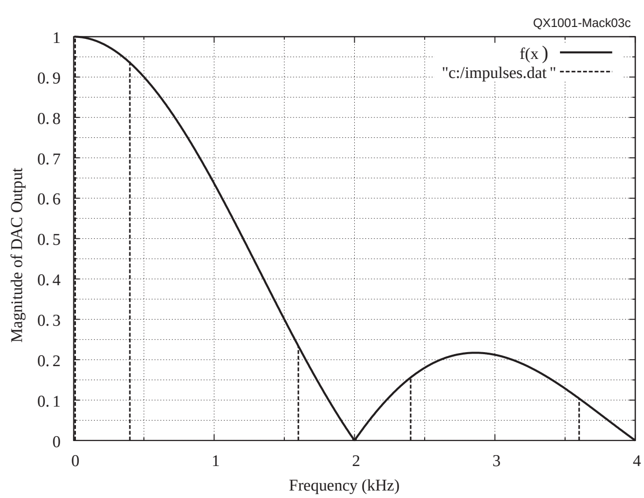
You will notice that the sinc waveform is essentially flat very near dc. If we sample




(A)

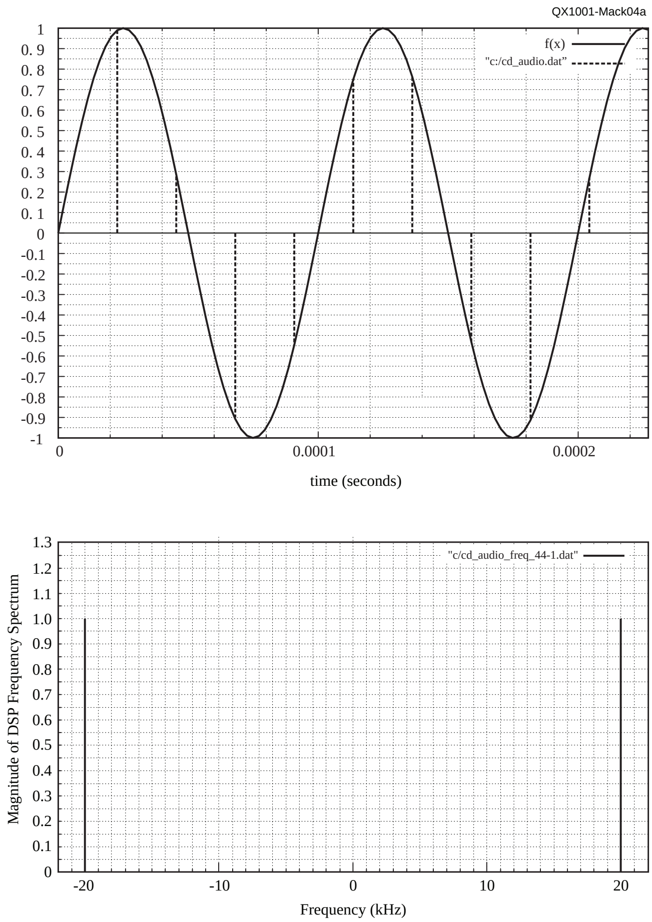



(C)

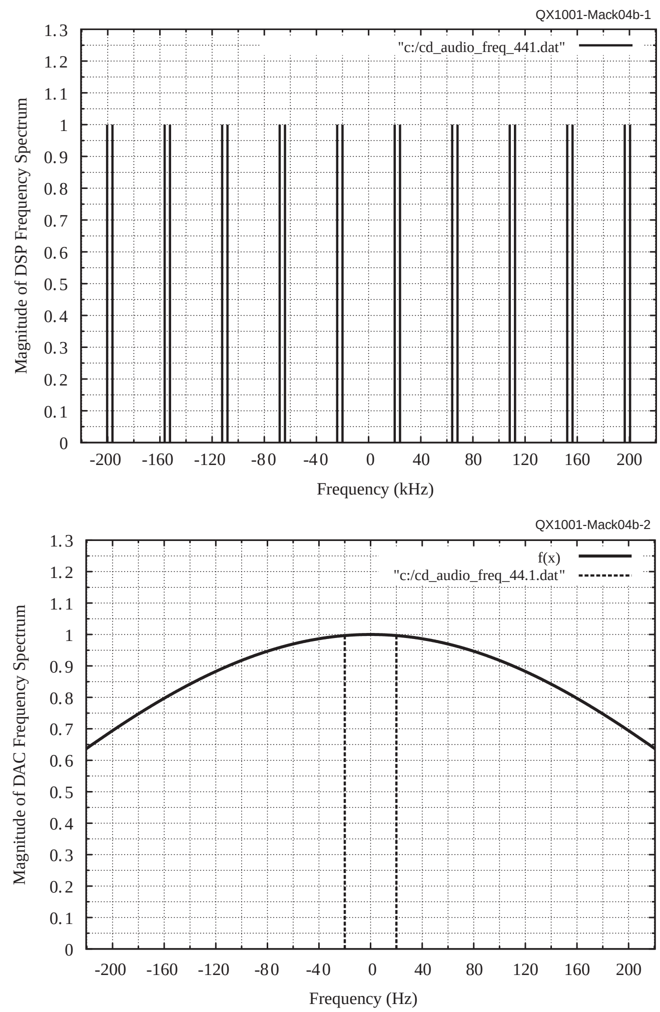



(B)

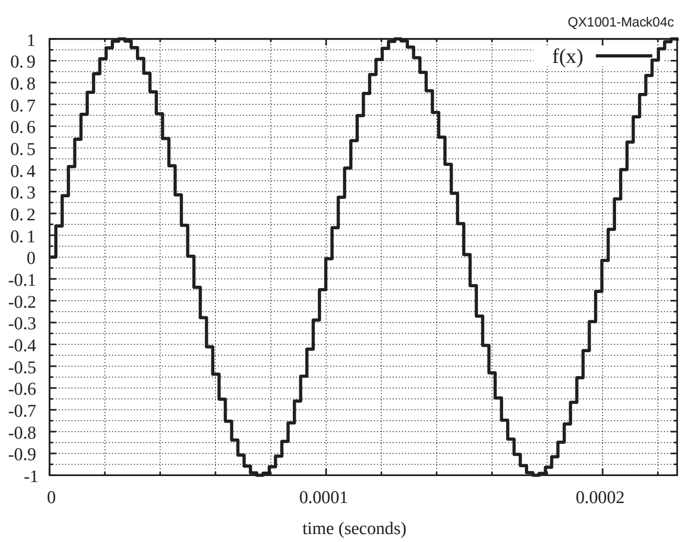
**Figure 3 — Part A is a graph of a 10 Hz sine wave and the DAC output of the 10 Hz sine when sampled at 2 kHz. The area between the two plots is extremely small, so very little energy appears at the DAC output at harmonics of the sample frequency. At B is a graph of a 400 Hz sine wave and the DAC output of the 400 Hz sine when sampled at 2 kHz. The area between the two plots is quite large, so there is substantial energy at the DAC output at harmonics of the sample frequency. Part C shows the DAC output spectrum for a 10 Hz signal and a 400 Hz signal when sampled at 2 kHz. The energy for the 10 Hz signal appears at full strength just to the right of the axis. The 400 Hz signal is down 0.5 dB from full strength. The 10 Hz signal is almost imperceptible in the second, third, and fourth Nyquist zones. The 400 Hz signal has substantial energy in each of the higher Nyquist zones. It is only 12 dB below the fundamental in the second Nyquist zone.**




(A)




(B)




(C)

**Figure 4 — Part A is a plot of the samples and original waveform of a 20 kHz CD audio tone when sampled at 44.1 kHz, along with the DSP spectrum. Part B shows a plot of the spectrum of a 20 kHz CD audio tone after inserting nine zero samples between each original sample. Note that the spectrum contains equal height tones that are two 20 kHz sidebands around each harmonic of the original 44.1 kHz sample frequency. The bottom graph shows the spectrum and sinc reduction after the top spectrum is passed through the interpolation filter. The strength of the output has improved from 3.9 dB down to only 0.18 dB down. In Part C we have a plot of the DAC output for the 20 kHz CD audio tone after a 10× interpolation filter.**

at a higher frequency, we can move the highest frequency of interest closer to the top part of the sinc waveform and lower the amount of compensation needed or ignore compensation altogether. In the case of CD audio, our highest frequency of interest is approximately 20 kHz. If we sample at 200 kHz, we move all of our information to the bottom 20% of the sinc function. Figure 2 shows the effects of using 192 kHz sampling (the current high end in common computer sound cards). The amount of compensation needed for excellent fidelity is much smaller. A good rule of thumb is that sampling at least ten times more than the required Nyquist rate will essentially eliminate the need for compensation.

We have looked at the math reason for sinc and seen a way to mitigate its effects, but why does it happen? Figure 3A shows a sine wave and its DAC waveform for 10 Hz with 2 kHz sampling. Figure 3B shows a sine wave and its DAC waveform for 400 Hz with 2 kHz sampling. Notice that the 10 Hz sampled waveform is a very close approximation to the sine wave. There is almost no variation from the actual sine wave. The 400 Hz waveform is another matter. Notice that there is significant area between the true sine wave and the DAC waveform. All of the area between the DAC waveform and the actual sine wave represents energy at frequencies other than 400 Hz. That energy shows up in the second, third and higher Nyquist zones. For the 10 Hz waveform, we see that the harmonic energy in the upper Nyquist zones falls under a very low part of the sinc function. This fits with our earlier observation that the size of the difference from the actual waveform is proportional to the energy in the higher Nyquist zones.

### Sample Rate Up Conversion

If we could take our CD audio at 44.1 kHz sample rate and increase the sample rate to 441 kHz (a 10× increase in sample frequency), we could eliminate a lot of the error in the signal compared to the original. One way to accomplish this sample rate conversion is to take two adjacent samples and draw a straight line between them. For our 10× increase in sample rate we would add 9 new samples with values between the two original samples. This is called interpolation, so the process of sample rate up conversion is called interpolation. This method does not really solve the problem. In the case of a sine wave at 20 kHz, we will have a sequence of sloped lines. The error is less but the error is still substantial.

We looked at sample rate down conversion (also called decimation or digital down conversion) in the Nov/Dec ’09 installment. The first application was to reduce the sample rate using a low pass filter, which made the first Nyquist zone smaller in frequency range. This allowed the frequency range to more closely match the capabilities of our digital signal processor. There is an inverse operation that uses the same principles to increase the sample rate, and therefore increase the size of the first Nyquist zone. Let’s look at our 20 kHz CD audio tone again.

I mentioned the limited frequency range of the DSP view of our signals last time, but did not go into detail. When we sample a waveform in the real world and convert it into the DSP world, we reduce the frequency domain from the real world infinite to the perfect, exact, and limited DSP frequency domain. The CD audio example has frequencies that extend from –22.05 kHz to +22.05 kHz and *only* those frequencies exist for the purposes of the math. The boundaries are entirely a consequence of the sample frequency. If we do a frequency domain plot (Discrete Fourier Transform) of our 20 kHz audio tone, we get a single line at –20 kHz and another at +20 kHz, and no other spectrum lines show up while we are working with DSP.

DSP interpolation uses the math to do a couple of transformations on the data. Figure 4A shows The input sine wave, the samples, and a frequency plot for a 10 sample sequence of the 20 kHz signal sampled at 44.1 kHz (22.7 µs between samples). Figure 4B shows what happens if we create a new sequence by adding nine zero samples between each of these original samples with 2.27 µs between each sample (441 kHz sample rate). Notice that the frequency plot now has equal height lines at –200.5 kHz, –196.4 kHz, –156.4 kHz, and so on, up to +200.5 kHz. These frequencies represent the fundamental at 20 kHz as well as the sum and difference between each harmonic of 44.1 kHz and the 20 kHz signal. It is a relatively simple matter, using DSP techniques, to create a brick wall low pass filter that will pass all frequencies below 22 kHz. The resulting sequence of samples is now 100 samples long, with each sample being a reasonable approximation to the 20 kHz sine wave. The process of adding the zero samples and low pass filter can be combined in one operation, and is called an interpolation filter.

Notice that our brick wall filter has to be pretty good, since it must pass 20 kHz and completely reject 24.1 kHz. A system with a signal so close to the top of the first Nyquist zone is a candidate for doing a 2× interpolation first, to move the first two sets of signals further apart and make the filter less critical. We haven’t looked

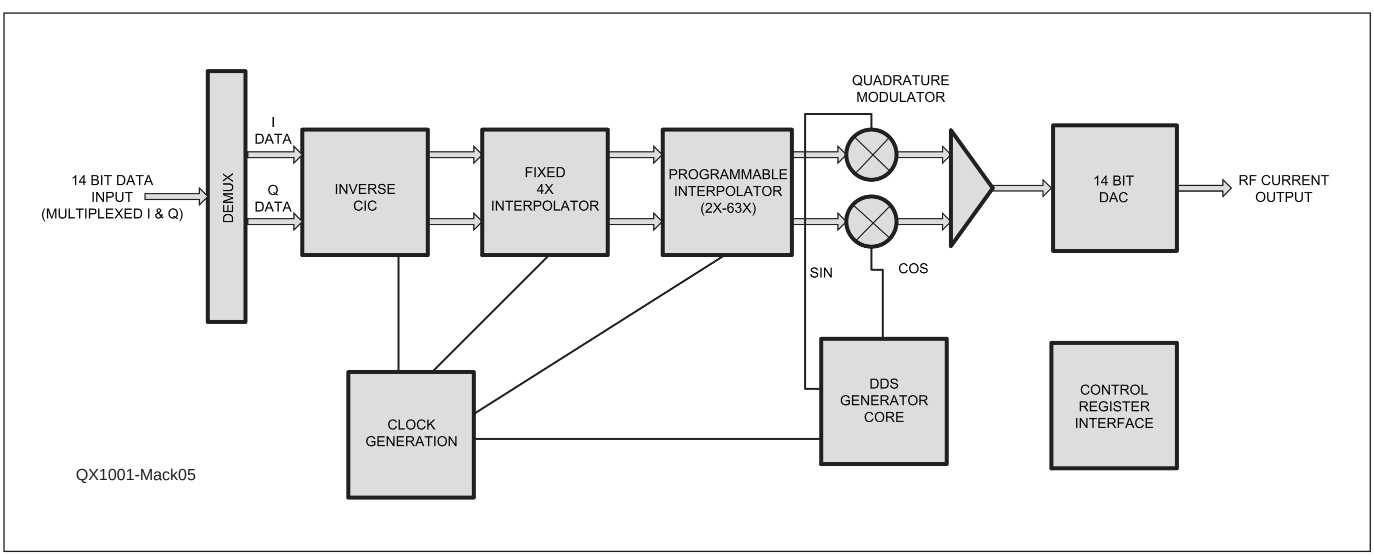



**Figure 5 — A simplified block diagram of the AD9857 transmitter IC showing two stages of interpolation and a final translation to RF using an image reject mixer. All functions occur in digital circuits prior to the output DAC. Down- load the data sheet from analog.com to see a more detailed block diagram.**

at filters in detail yet, but the ratio of filter bandwidth to total bandwidth affects the computer resources. A filter with a ratio of 0.5 requires significantly less CPU power than a filter with a 0.1 ratio. This is similar to the analog world, where a 500 Hz CW filter requires a lot more crystals for sharp skirts than a 6 kHz AM filter.

We looked at band pass decimation in the Nov/Dec ’09 issue, where we were able to increase the ratio of our signal bandwidth to total DSP bandwidth and also change the frequency of the signals. Again, the inverse function works for us. It is very easy to start with audio from 300 Hz to 3 kHz and create a DSB suppressed carrier signal in DSP at 200 kHz, for example, with an 800 kHz sample rate. Then we can create a DSP brick wall band pass filter to create either upper sideband or lower sideband by filtering. Now we can use the same zero sample insertion method we talked about above to raise the sample rate. We could insert four zero samples between each original sample to increase the sample rate to 4.0 MHz. This would result in new transmitter signals at 200 kHz, 600 kHz, 1000 kHz, 1400 kHz, and 1800 kHz. We can create a band pass filter centered on any one of these signals to simultaneously increase the sample rate to avoid sinc issues as well as translate the SSB signal to a new frequency. An interesting example would take the new 200 kHz signal and create a 2× interpolation filter to increase the sample rate again to 8.0 MHz. The band pass interpolation filter can be centered at 3.8 MHz to move our original 200 kHz SSB signal to the 75 m band without any analog operations, other than the final DAC step with its reconstruction filter. Another interesting example would band pass select the 1000 kHz signal and use a 3× interpolator to generate an SSB signal at 5.0 MHz for the 60m band.

The interpolation and digital filtering for the higher frequencies is frequently performed by an FPGA in modern systems. A large FPGA can contain the logic to insert the zero samples and perform the multiply/accumulate functions of the digital filtering. The interpolation is a fixed function and does not lend itself to the general purpose capabilities of a DSP computer.

The AD9756 and AD9857 are interesting ICs from Analog Devices. These parts implement the core of a digital radio transmitter. Both parts implement a 200 MHz sample rate system, but the AD9856 has a 12 bit data path and the AD9857 has a 14 bit data path. The block diagram for the parts is shown in Figure 5. The intent of these parts is that you will create a baseband version of your transmitter signal with both *I* and *Q* versions. The sample rate of the baseband signal is then interpolated from perhaps 1 MHz or 10 MHz up to the final sample rate of the system (potentially as high as 200 MHz). The interpolation is the low pass baseband variety rather than the band pass up conversion type. The chip then uses the *I* and *Q* output of a direct digital synthesizer to create an image reject mixer that operates entirely on digital samples. The result is a transmitter signal that is entirely digital until the DAC connects to the antenna. This would be a QRPP signal, since the DAC can directly generate about 10 dBm at its output. The IC actually has three modes. The first is the transmitter mode just described. It can also be used as a DDS single tone generator for applications like a local oscillator. The third mode just operates as a baseband interpolation filter and DAC to raise the sample rate.

### Future Issues

Thanksgiving and Christmas breaks are coming, so my intention is to use the extra time to create experimental boards for both the AD9857 and the AD9864. The goal is to create an SSB transmitter for 160 m through 6 m (including 4 m for the UK) where the generation is done entirely in the digital world, with the exception of the final power amplifier. The AD9864 is billed as a 10 MHz to 300 MHz digital receiver IC. It will be interesting to see if it is possible to rearrange the connections to create a companion receiver that will receive the same frequency range. This IC first converts the input to a frequency between 1.6 MHz and 3 MHz, so it is not a true “connect to antenna” software defined radio IC. It samples the output of the mixer with a 24 bit ADC and processes it in a set of decimation filters to drop the frequency down to the 200 kHz range.

Another task is to develop some software to demonstrate both decimation and interpolation. I have finally realized that I cannot avoid having a *Linux* development system. So much of the resources for the Stamp are developed only on *Linux* and must be adjusted to work with *Windows* based tools. My system isn’t all the way built yet, so that is another task for the holidays. QEX

**Mike Hamel, WO1U**
5 Ridge Rd, Essex Junction, VT 05452; wo1u@arrl.net

**Dave Cripe, NMØS**
118 Hilltop Dr NE, Mt Vernon, IA 52314; nm0s@arrl.net

# Tech Notes

## More on Phase Controlled Differential Drive (or) Outphasing Modulation of High Efficiency Amplifiers

It has bugged me for some time that there must be a better way to modulate a high efficiency switch-mode RF amplifier than by using a modulated power supply. Also, since there are numerous and very old references to linear amplification using non-linear components (LINC), it is surprising how little of this kind of work is being done by Amateur Radio operators. It was this thinking that prompted my article “Phase Controlled Differential Drive for EER Amplifiers” in the Sep/Oct 2009 issue of *QEX*.[1] One goal of the article was to start some discussion on the subject, and also to see if I could learn of some better techniques from this discussion. I have since received a lot of comments and notable corrections.

Shortly after publication of the article, Dr. Fritz Raab, W1FR, alerted me to the correct term for the technique (outphasing), and cited a number of great references including the 1935 Chireix patent of outphasing amplifiers. Dr. Raab also alerted me to the fact that a fully differential phase modulator must be used to avoid phase modulation appearing at the output. While reading these references and searching for more about the technique, I found another reference to using the outphasing technique to drive a pair of identical Class E stages and combining the output. R. A. Beltran, F. H. Raab and A. Velazquez presented “HF Outphasing Transmitter using Class-E Power Amplifiers” at the IEEE Microwave Theory and Techniques Society 2009 *International Microwave Symposium*, in June 2009.[2]

[1]Notes appear on page 36.

I then received a letter from David Cripe, NMØS, pointing out many of the issues I had run into, and also alerting me to the original references on these techniques. In discussion with Dave about the amplifier in the article, he also described an effective dual Class E outphasing configuration that was simpler than the circuit of the Beltran, Raab and Velasquez article. This design maintains high efficiency across full modulation more effectively than the CMCD configuration of my *QEX* article.

### Comments From NMØS

It is well-known that when the outputs of two power amplifiers are directly combined, a phase difference in their drive signals will result in a current between them, causing an asymmetry in their respective voltage and

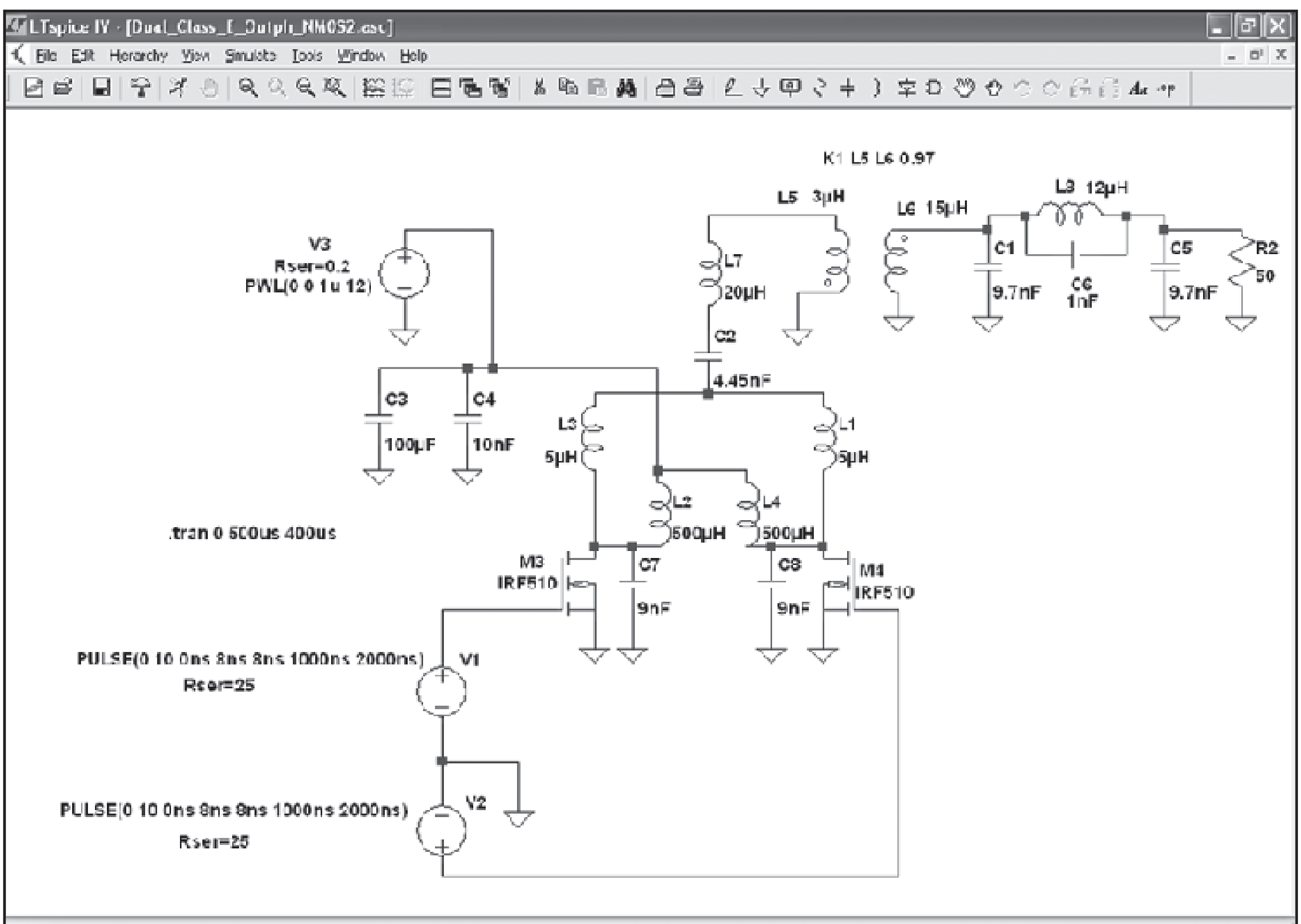



**Figure 1 — This screen shot shows the schematic diagram of the outphase modulated amplifier from the *LTSpice* schematic capture program.**

current waveforms. Chireix, the inventor of outphasing, discovered an output network that would maintain near-optimal tuning of the two PA sections as the differential phase between them was varied.

Particular care must be taken in the design of highly efficient switching power amplifiers operating Class-E or Class-F that are intended for outphasing modulation. The mutual load-pulling of the outputs of the two PAs under outphasing can cause their operation to become extremely inefficient, possibly resulting in their failure. The Beltran, Raab and Velasquez circuit is noteworthy in that it successfully applies a variation on the concepts of Chireix to allow Class-E power amplifiers to be outphased, yielding a circuit with excellent efficiency throughout its modulation range.

A much simpler solution to the problem of combining Class-E PAs to permit outphasing modulation actually comes from the world of high powered AM and Shortwave broadcast transmitters. The outputs of two, identical (not outphased) transmitters are commonly combined through two, equal inductors. This permits a degree of imbalance or phase error between their outputs to exist without damaging either transmitter. To do the same with a pair of Class-E power amplifiers serendipitously results in performance very similar to that of the Beltran, Raab and Velasquez circuit, with much less complexity.

**WO1U Models the Design**

Figure 1 shows the schematic diagram of a 500 kHz version of the NMØS amplifier configuration, as created by an *LTSpice* model.[3] Figures 2 and 3 each show a screen shot of the simulation output at 20% and full output level respectively. Note that the efficiency at twenty percent output is 65% and at full output it is 96%. In a real amplifier it is difficult to achieve such high actual efficiency due to non ideal characteristics of real components.

Output level is changed by adjusting the delay in the PULSE *Spice* directive of V1 or V2. It should be noted that in this amplifier, full output is achieved when the drive signals are in phase and minimum output occurs when drive is 180° out of phase. Care must be taken to ensure that the second harmonic is well filtered, since the minimum output point at the desired frequency is where the maximum second harmonic level occurs. A third order Cauer low pass filter is shown. This filter was designed to have a notch in the stop band at the second harmonic frequency, 1 MHz.

This amplifier can be scaled to ham band frequencies by choosing component values with the same reactance in the frequency range of interest. The model uses components built into the *LTSpice* libraries and no customization was done. I cannot say enough good things about the *LTSpice* design tool; it is free and easy to use and available for download from the Linear Technology Web site.

In summary, a viable design is presented here that addresses the shortcomings of the CMCD outphased amplifier described in the Sep/Oct 2009 *QEX* article. The configuration was modeled and results are presented. The *LTSpice* model file is available for download on the *QEX* files Web page.[4]

I would like to thank Dr. Fritz Raab, W1FR, and Dave Cripe, NMØS, for their valued input and encouragement. This will make a great winter project, and I hope this Tech Note will inspire others to experiment in this area.

**Notes**

[1]Mike Hamel, WO1U, "Phase Controlled Differential Drive for EER Amplifiers," Sep/Oct 2009 *QEX*, pp 31-34.

[2]R. A. Beltran, F. H. Raab, A. Velazquez, "HF Outphasing Transmitter using Class-E Power Amplifiers"; *International Microwave Symposium,* June 2009, pp 757-760; (ISBN: 978-1-4244-2804-5).

[3]Linear Technology offers *LTSpice* for free download. This *Spice III* circuit simulator includes schematic capture and waveform viewer enhancements. Go to **www.linear.com/designtools/software/.**

[4]The author's *LTSpice* circuit simulation files are available for download from the ARRL *QEX* Web site. Go to **www.arrl.org/qexfiles** and look for **1x10_Hamel.zip**

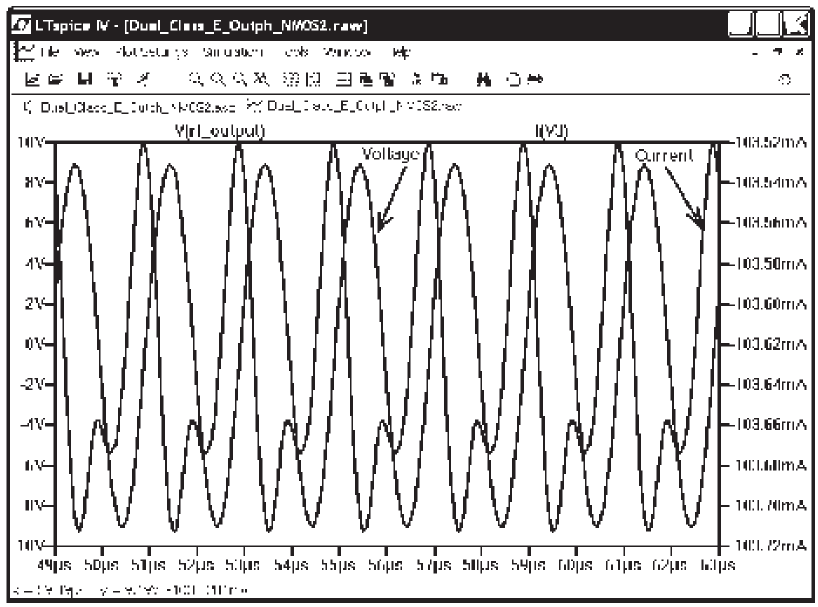



**Figure 2 — Here is an *LTSpice* screen shot of the amplifier waveforms at 20% output level. One trace is the RF carrier, and the other is the dc supply current.**

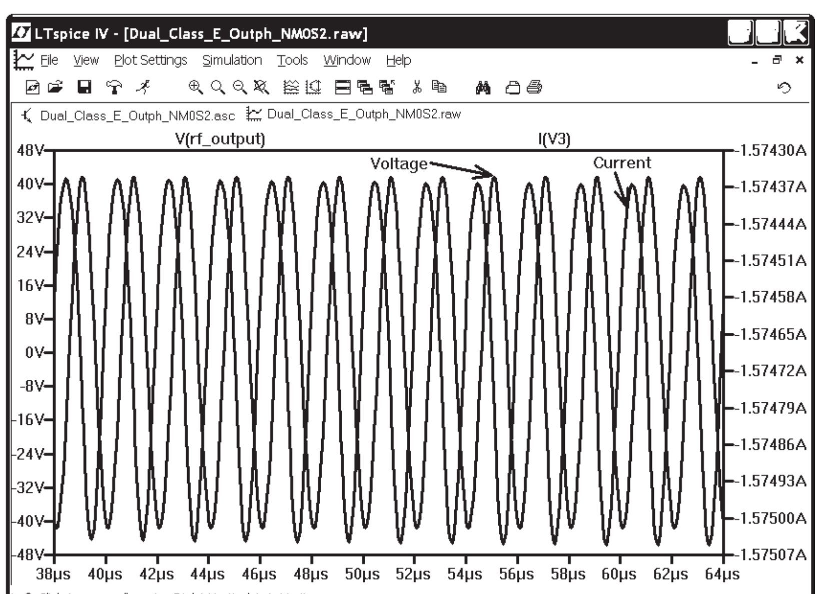



**Figure 3 — This is an *LTSpice* screen shot of the amplifier waveforms at full output.** QEX

# Letters to the Editor

### Experimental Determination of Ground System Performance for HF Verticals — Part 5 (Jul/Aug 2009)

**Dear Larry,**

The work of Rudy Severns, N6LF — the latest in the Jul/Aug 2009 issue of *QEX* — needs to be made available to a much wider audience. I ‘d like to suggest that his series be summarized for the *QST* audience which, I believe, is much less technically knowledgeable than that of *QEX*. Better still, his findings should be incorporated in the next edition of *The ARRL Antenna Book*.

The piece by Bob Zavrel, W7SX, in the same issue of *QEX* is a beautiful presentation! For those not as technically inclined: read, and believe the Conclusions on page 33. They effectively refute the mythology and mumbo jumbo heard on the air with increasing frequency. It, too, belongs in the *Antenna Book*.

*— Very 73, Jim Olsen Jr, 5905 Landon Ln, Bethesda, MD 20817;* **w3kmn@aol.co**

**Hi Jim,**

Thanks for the kind words. I hope this information helps people.

*— 73, Rudy Severns, N6LF, PO Box 589, Cottage Grove, OR 97424;* **n6lf@arrl.net**

**Hi Jim,**

Thank you for very much for your comments. ARRL Technical Editor Joel Hallas, W1ZR, has been talking with N6LF about a summary of his *QEX* articles for publication in *QST*.

*— 73, Larry Wolfgang, WR1B,* QEX *Editor;* **lwolfgang@arrl.org**

### Maximizing Radiation Resistance in Vertical Antennas (Jul/Aug 2009)

**Dear Mr. Zavrel,**

The Jul/Aug issue of *QEX* arrived today, and I was just beginning to peruse your article when I noticed repeated references to a publication by “John Krauss.” The classic *Antennas* by John D. Kraus has for many years been my Bible for antenna engineering.

Respectfully, unless you are referring to a newcomer of whom I am unaware, the spelling of John’s surname remains “Kraus.”

I look forward to finishing your article.

*— 73, Don T. Batson, W4TTQ, 8442 Antero Dr, Austin, TX 78759;* **w4ttq@arrl.net**

**Hi Don,**

Thank you for your comments. I hope you enjoyed reading the entire article. Of course, Bob and I both know how to spell Dr. Kraus’s last name, and I sincerely apologize for that error! I even had his book out of our Technical Library to check that referenced text. Ten lashes with a wet antenna wire for both of us.

*— 73, Larry, WR1B*

### Letters, ResCad.exe Program Files Sep/Oct 2009)

**Hi Larry,**

Thanks for the tip regarding *ResCad* in the Sep/Oct issue of *QEX*. This will be a useful addition to any electronic toolbox. Searching the Web, I found that this little package normally includes a short text file explaining its use. This text file was not part of the package that I downloaded from **www.arrl.org/qexfiles**. The program’s author permits the free distribution of the program but suggests that the text file be included in any distribution. The full distribution, including the text file, can be downloaded from **www.armory.com/~rstevew/Public/Software/**.

I found an interesting comment from Terry Pinnel, a hobbyist in the UK who points out that *ResCad* can be applied to capacitors too, if serial and parallel modes are interchanged. You can read Terry’s comments at **http://sci.tech-archive.net/Archive/sci.electronics.basics/2004-11/1310.html**.

I look forward to each issue of *QEX*.

*— 73, Dave Green, VE3TLY, 410 Hamilton Ave, Ottawa, ON K1Y 1E1, Canada;* **ve3tly.gu9z@ncf.ca**

**Hi Dave,**

Thank you for that additional information about *Resistor CAD*. The Web site given in the Sep/Oct Letters column did not include the RESCADREADME.TXT file. I have now downloaded the complete package and placed the new **9x09_rescad.zip** file on the ARRL *QEX* Files Web site.

*— 73, Larry, WR1B*

### Some Thoughts on Crystal Parameter Measurement (Letters, Sep/Oct 2008)

**Editor, *QEX* Magazine,**

Jim Koehler, VE5FP, was kind enough to point out a rather serious typographical error in the crystal resonance equations in my letter published in the Sep/Oct 2008 issue of *QEX*. The parallel-resonant frequencies of maximum impedance ($f_{pz}$) and of zero phase ($f_{p\phi}$) are symmetrically above and below the nominal parallel-resonant frequency, not the series-resonant frequency indicated by the equations as published.

For reference, here is the complete set of equations, hopefully correct this time!

The motional series-resonant frequency is

$$f_{sm} = 1 / \left(2\pi\sqrt{L_m C_m}\right)$$

where $L_m$ and $C_m$ are the motional inductance and capacitance of the standard crystal model (Figure 1).

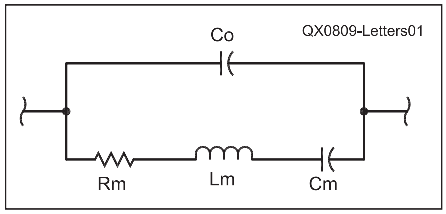



**Figure 1—Equivalent circuit for a quartz crystal.**

Including the effects of the motional resistance $R_m$ and the shunt capacitance $C_0$, the series-resonant frequencies of minimum impedance ($f_{sz}$) and of zero phase ($f_{s\phi}$) are

$$f_{sz} = f_{sm} - \Delta$$

$$f_{sz} = f_{sm} - \Delta$$

where

$$\Delta = f_{sm}(1/2Q^2)(C_0 / C_m)$$

$$Q = 2\pi \cdot f_{sm} L_m / R_m$$

The nominal parallel resonant frequency (what the parallel resonant frequency would be if $R_m$ were zero ohms) is:

$$f_{pm} = f_{sm}\sqrt{1 + C_m / C_0}$$
$$\approx f_{sm}\left(1 + 0.5 C_m / C_0\right)$$

The corrected equations for the parallel-resonant frequencies that include the effect of $R_m$ are

$$f_{pz} = f_{pm} + \Delta$$

$$f_{p\varphi} = f_{pm} - \Delta$$

where $f_{pz}$ is the frequency of maximum impedance and $f_{p\phi}$ is the parallel-resonant frequency of zero phase.

One common measurement technique is to place the crystal under test in series between a signal source and measuring device. The source and load impedance are typically 50 Ω, however lower impedances are used in some test systems for better measurement accuracy. The frequency of maximum transmission of such a pi-network circuit is not equal to any of the three previous definitions of series-resonant frequency. It is

$$f_{\pi} = f_{sm} - \Delta\left(1 + 4R_T / R_m\right)$$

Where $R_T$ is the source and load resistance (assumed equal) in the fixture.

It should be mentioned that the factor Δ in the above equations is an approximation that is most accurate when the ratio of the reactance of $C_0$ to $R_m$ is as high as possible. For most crystals the approximation is very good. Some high-frequency overtone crystals have a low-enough Q, however, that the motional resistance may approach the reactance of the shunt capacitance, which degrades the accuracy of the approximation.

*— 73, Alan Bloom, N1AL, 1578 Los Alamos Rd, Santa Rosa, CA 95409;* **n1al@arrl.net**

**Hi Alan,**

Thank you for sending the detailed information about the equations from your earlier letter.

*— 73, Larry, WR1B*

QEX

# Upcoming Conferences

## 2010 Southeastern VHF Society Conference

April 23-24, 2010
Morehead State University in Morehead Kentucky

The Southeastern VHF Society is calling for the submission of papers and presentations for the upcoming 14th Annual Southeastern VHF Society Conference to be held at Morehead State Univeristy in Morehead, KY on April 23rd and 24th, 2010. Papers and presentations are solicited on both the technical and operational aspects of VHF, UHF and Microwave weak signal Amateur Radio. Some suggested areas of interest are:

- Transmitters
- Receivers
- Transverters
- RF Power Amplifiers
- RF Low Noise Pre Amplifiers
- Antennas
- Construction Projects
- Test Equipment And Station Accessories
- Station Design And Construction
- Contesting
- Roving
- DXpeditions
- EME
- Propagation (Sporadic E, Meteor Scatter, Troposphere Ducting, etc.)
- Digital Modes (WSJT, etc.)
- Digital Signal Processing (DSP)
- Software Defined Radio (SDR)
- Amateur Satellites
- Amateur Television

In general papers and presentations on non weak signal related topics such as FM repeaters and packet will not be accepted but exceptions may be made if the topic is related to weak signal. For example, a paper or presentation on the use of APRS to track rovers during contests would be considered.

The deadline for the submission of papers and presentations is February 5, 2010. All submissions for the proceedings should be in Microsoft *Word* (.doc) format. Submissions for presentation at the conference should be in Microsoft *PowerPoint* (.ppt) format, and delivered on either a USB memory stick or CDROM, or posted for download on a Web site of your choice.

Pages are 8½ by 11 inches with a 1 inch margin on the bottom and ¾ inch margin on the other three sides. All text, drawings, photos, etc. should be black and white only (no color).

Please indicate when you submit your paper or presentation if you plan to attend the conference and present there, or if you are submitting just for publication. Papers and presentations will be published in bound proceedings by the ARRL. Send all questions, comments and submissions to the program chair, Robin Midgett, K4IDC, via **K4IDC@comcast.net**.

**Upcoming Conferences continued on page 40.**

# 2009 *QEX* Index

## Feature Articles



## About the Cover

QEX

# Upcoming Conferences

## Society of Amateur Radio Astronomers

July 4-7, 2010
National Radio Astronomy Observatory
Green Bank, West Virginia

**Call for Papers: 2010 Annual Meeting**

The Society of Amateur Radio Astronomers (SARA) hereby solicits papers for presentation at its 2010 Annual Meeting and Technical Conference, to be held July 4 through 7, 2010, at the National Radio Astronomy Observatory (NRAO), Green Bank West Virginia. Papers on radio astronomy hardware, software, education, research strategies, and philosophy are welcome.

SARA members or supporters wishing to present a paper should email a letter of intent, including a proposed title and informal abstract or outline (not to exceed 100 words) to the SARA vice president at **vicepres@radio-astronomy.org**, no later than 1 March 2010. Be sure to include your full name, affiliation, postal address, and e-mail address, and indicate your willingness to attend the conference to present your paper. Submitters will receive an e-mail response, typically within one week, along with a request to proceed to the next stage, if the proposal is consistent with the planned program. A formal *Proceedings* book will be published in conjunction with this meeting. Papers will be peer-reviewed by a panel of SARA members with appropriate professional expertise and academic credentials. First-draft manuscripts must be received no later than 1 April 2010, with feedback, acceptance, or rejection e-mails to be sent within two weeks thereafter. Upon final editing of accepted papers, camera-ready copy will be due not later than 1 May 2010. Due to printer's deadlines, manuscripts received after that deadline will not make it into the *Proceedings*. Instructions for preparation and submission of final manuscripts appear in a "Guidelines for Submitting Papers" document on the SARA Web site at **www.radio-astronomy.org/node/43**.

The last three year's *Proceedings* were a landmark accomplishment for our organization. Please help the Society of Amateur Radio Astronomers to make the 2010 edition even better!

## 44th Annual Central States VHF Society Conference

July 22–24, 2010,
Doubletree Inn
Bridgeton, Missouri

Presentations aren't necessarily technical — they cover the breadth of the VHF/UHF ham radio hobby. Highlights in past years have been demonstrations of Software Defined Radio and LASER Communication beyond line-of-sight. Presentations generally vary from 15 to 45 minutes and step you through the highlights of the topic at hand, with complete texts published as articles in the proceedings. Further details and a Call for Papers were not available at *QEX* press time. In past years, papers for presentation at the Conference and inclusion in the *Proceedings* book as well as posters for table-top display during the Conference have been solicited.

# THANK YOU D-STAR DEVELOPERS

## USER DEVELOPED D-STAR ACHIEVEMENTS

***D-STAR Monitor***
Software to report D-STAR activity to the *Last Heard Report* at www.dstarusers.org

***D-STAR TNC***
Software for translating D-PRS data to basic APRS® messages for APRS® position maps.

***µSmartDigi™ D-Gate™ D-STAR Gateway***
Compact, portable TNC designed to gateway position packets between a D-STAR digital network and a conventional analog APRS® network via a D-STAR radio and a conventional radio.

***DStarQuery***
Software that works in tandem with javAPRSSrvr or D-PRS interface to monitor the D-STAR serial data port.

***D*Chat***
Software for keyboard to keyboard communication.

***D-RATS***
Software package to provide data communication capabilities with D-STAR radios.
Supports text messaging, file transfers, position reporting, mapping, and much more!

***DPLUS***
Software package that extends features not natively available on the Icom Gateway software.

***DV Dongle***
Hardware integrating computers into the D-STAR network for communications.

***D-STAR Calculator***
An online resource designed to simplify D-STAR operations.
By selecting a few options, the calculator tells you what you need to program into your radio.
While we can't program your radio, stay tuned for future enhancements!

www.dstarinfo.com/applications/applications.htm

D-PRS™

ICOM®