

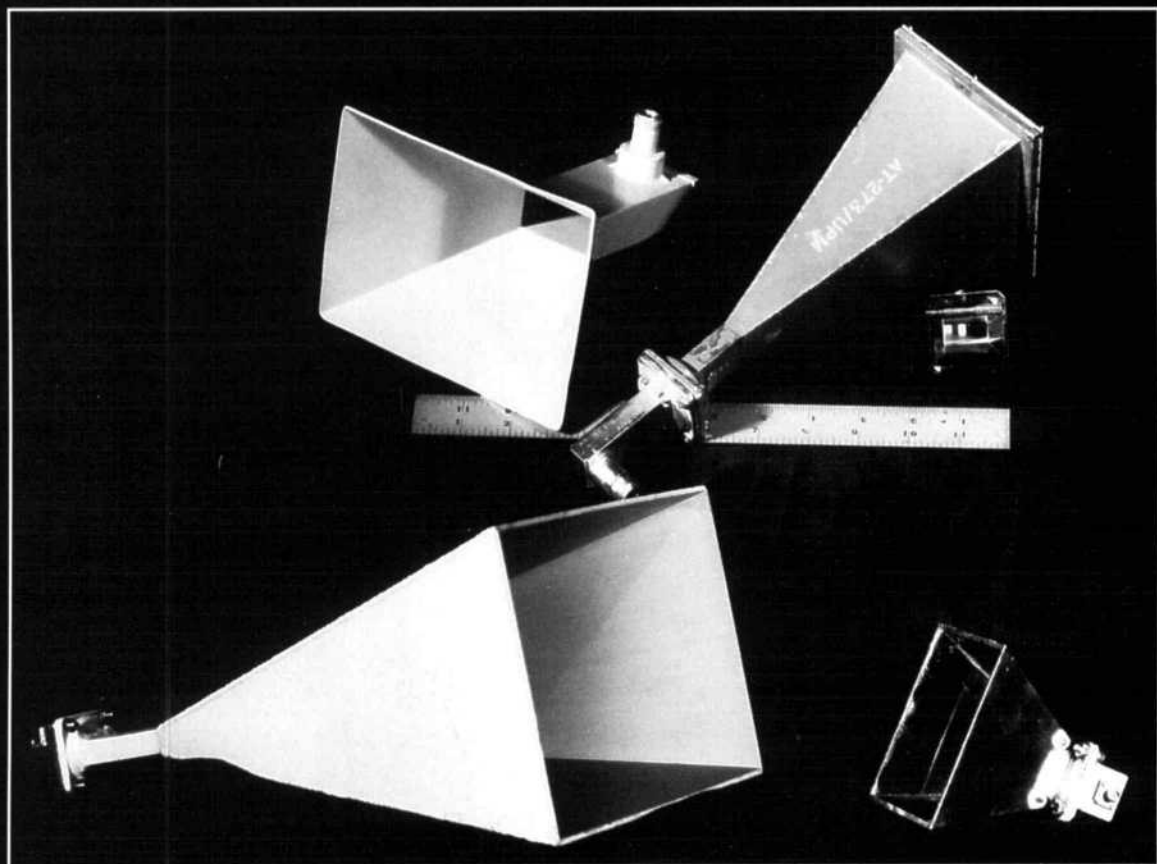
QEX

\$1.75



ARRL Experimenter's Exchange

September 1994



Horn in on the microwave bands!

QEX: The ARRL
Experimenter's Exchange
American Radio Relay League
225 Main Street
Newington, CT USA 06111

QEX

QEX (ISSN: 0886-8093 USPS 011-424) is published monthly by the American Radio Relay League, Newington, CT USA.

Second-class postage paid at Hartford, Connecticut and additional mailing offices.

David Sumner, K1ZZ
Publisher

Jon Bloom, KE3Z
Editor

Lori Weinberg
Assistant Editor

Harold Price, NK6K
Zack Lau, KH6CP
Contributing Editors

Production Department

Mark J. Wilson, AA2Z
Publications Manager

Michelle Bloom, WB1ENT
Production Supervisor

Sue Fagan
Graphic Design Supervisor

Dianna Roy
Senior Production Assistant

Joe Shea
Production Assistant

Advertising Information Contact:

Brad Thomas, KC1EX, Advertising Manager
American Radio Relay League
203-667-2494 direct
203-666-1541 ARRL
203-665-7531 fax

Circulation Department

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

Offices

225 Main St, Newington, CT 06111-1494
USA
Telephone: 203-666-1541
Telex: 650215-5052 MCI
FAX: 203-665-7531 (24 hour direct line)
Electronic Mail: MCIMAILID: 215-5052
Internet: qex@arrl.org

Subscription rate for 12 issues:

In the US: ARRL Member \$12,
nonmember \$24;

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

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

QEX subscription orders, changes of address, and reports of missing or damaged copies may be marked: QEX Circulation. Postmaster: Form 3579 requested. Send change of address to: American Radio Relay League, 225 Main St, Newington, CT 06111-1494.

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

Copyright © 1994 by the American Radio Relay League Inc. Material may be excerpted from QEX without prior permission provided that the original contributor is credited, and QEX is identified as the source.



About the Cover:

Microwave antennas don't have to be difficult, as N1BWT shows us.

ISSUE
NO.
151



Features

3 Practical Microwave Antennas

Part 1 of 3

By Paul Wade, N1BWT

12 Wavelet Compression for Image Transmission Through Bandlimited Channels

By A. Langi, VE4ARM, and W. Kinser, VE4WK

22 Negative Frequencies and Complex Signals

By Jon Bloom, KE3Z

Columns

28 RF

By Zack Lau, KH6CP/1

September 1994 QEX Advertising Index

American Radio Relay League: 32
Communications Specialists
Inc: 11
Down East Microwave: 11
L.L. Grace: Cov II
LUCAS Radio/Kangaroo Tabor

Software: 27
PacComm Packet Systems: Cov III
Tucson Amateur Packet Radio Corp: 32
Yaesu: Cov IV
Z Domain Technologies, Inc: 32



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

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

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

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

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

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

Officers

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

Executive Vice President: DAVID SUMNER, K1ZZ

Purpose of QEX:

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

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

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

Any opinions expressed in QEX are those of the authors, not necessarily those of the editor or the League. While we attempt to ensure that all articles are technically valid, authors are expected to defend their own material. Products mentioned in the text are included for your information; no endorsement is implied. The information is believed to be correct, but readers are cautioned to verify availability of the product before sending money to the vendor.

Empirically Speaking

ARRL Technical Workshops

In the past two years, ARRL's Educational Activities Department (EAD) has sponsored several technical workshops, beginning with an "Introduction to Digital Signal Processing," which we have run four times. Additional workshops scheduled for 1994 are "Electromagnetic Interference," which will be held September 30, in Boxboro, Massachusetts, in conjunction with the ARRL New England Division Convention, and "Computer-Aided Design of HF Antennas," scheduled for October 21, in Concord, California, at the ARRL Pacific Division Convention (Pacificon). (Contact EAD at ARRL Headquarters for further information about these workshops.)

The purposes of these events are several. While they don't take the place of quality published material (magazine articles and books) as tutorials, in-person workshops provide an opportunity to delve intensely into a subject, taught by an instructor who is available to answer questions and give further detail about those aspects of the subject that are unclear. In presenting these workshops, our goal is to foster increased activity in the topics covered. One of the general subject areas is that of advanced technology. We hope to spur people to do more work in fields that are, perhaps, underused by amateurs, thus advancing amateur technology in general. (Particularly if the work is then published!) Another goal is to increase amateurs' knowledge of technical subjects that are important to the future of Amateur Radio—subjects such as EMI control. We also use these workshops as opportunities to find out what technical subjects are of interest to our members, so we can better serve their information needs.

Which brings us to the point. We're presently in the initial planning stages for 1995. An as-yet-unspecified number of workshops will be sponsored next year. We'd like to know what topics you would like to see covered. A topic may be "high-tech," or it may not. But it should be one which is broad enough to make for a 5- or 6-hour workshop, and it should be rel-

evant to Amateur Radio. Which of the suggested topics we will eventually cover depends on a number of factors, including the availability of qualified instructors for the workshops. But the first step is to find out what topics you are interested in. Send your suggestions to: Rosalie White, Educational Activities Department, ARRL, 225 Main Street, Newington, CT 06111, or email them to rwhite@arrl.org (Internet). The more detail you can provide about the material you think should be covered, the better. We're also looking for good sites at which to hold workshops, so you may want to see if the folks who are running conventions in your Division during 1995 are interested in hosting an ARRL technical workshop next year.

This Month in QEX

Are you thinking of getting on the microwave bands but nervous about your ability to construct a usable microwave antenna? There's no need to be; it's easier than you think to build "Practical Microwave Antennas," as Paul Wade, N1BWT, shows us in part 1 of a three-part series.

Some of the most interesting technical material around appears in the various amateur conference proceedings. This month, we reprint a paper from this year's *ARRL Digital Communications Conference*: "Wavelet Compression for Image Transmission Through Bandlimited Channels," by A. Langi, VE4ARM, and W. Kinsner, VE4WK. It's an interesting paper in its own right, and may introduce the proceedings to those who have never seen them.

"Negative Frequencies and Complex Signals" are not well understood by many amateurs. They are important subtopics of signal processing, however, especially in DSP. Your editor tries to clear up some of the confusion in this article.

In this month's "RF" column, Zack Lau, KH6CP/1, provides a design for a 5760-MHz preamp that gives impressive noise-figure performance and gain adequate to make the preamp the factor controlling the receiving system's sensitivity.—KE3Z, email: jbloom@arrl.org (Internet)

Practical Microwave Antennas

Part 1-Antenna fundamentals and horn antennas

by Paul Wade, N1BWT

Antenna gain is essential for microwave communication, and since it helps both transmitting and receiving, it is doubly valuable. Practical microwave antennas provide high gain within the range of amateur fabrication skills and budgets.

Three types of microwave antennas meet these criteria: horns, lenses and dishes. Horns are simple, foolproof and easy to build; a 10-GHz horn with 17 dB of gain fits in the palm of a hand. Metal-plate lenses are easy to build, light in weight and noncritical to adjust.¹ Finally, dishes can provide extremely high gain; a 2-foot dish at 10 GHz has more than 30 dB of gain, and much larger dishes are available.

These high gains are only achievable if the antennas are properly implemented. I will try to explain the fundamentals using pictures and graphics as an aid to understanding. In addition, a computer program, *HDL_ANT*, is available for the difficult calculations and details. In this

first of three parts, I'll review some basic antenna terminology and concepts and discuss horn antennas. Part 2 will treat dish antennas, and in Part 3 I'll present metal-lens antennas and discuss the microwave antenna measurements needed to verify antenna performance.

Antenna Basics

Before we talk about specific microwave antennas, there are a few common terms that must be defined and explained:

Aperture

The aperture of an antenna is the area that captures energy from a passing radio wave. For a dish antenna, it is not surprising that the aperture is the size of the reflector, and for a horn, the aperture is the area of the mouth of the horn. Wire antennas are not so simple—a thin dipole has almost no area, but its aperture is roughly an ellipse with an area of about $0.13\lambda^2$. Yagi-Uda antennas have even larger apertures.²

Gain

The hypothetical isotropic antenna is a point source that radiates equally in

all directions. Any real antenna will radiate more energy in some directions than in others. Since the antenna cannot create energy, the total power radiated is the same as that of an isotropic antenna driven from the same transmitter; in some directions it radiates more energy than an isotropic antenna, so in others it must radiate less energy. The gain of an antenna in a given direction is the amount of energy radiated in that direction compared to the energy an isotropic antenna would radiate in the same direction when driven with the same input power. Usually we are only interested in the maximum gain—in the direction in which the antenna is radiating most of the power.

An antenna with a large aperture has more gain than a smaller one; just as it captures more energy from a passing radio wave, it also radiates more energy in that direction. Gain may be calculated as:

$$G_{dB} = 10 \log_{10} \left(\eta \cdot \frac{4\pi}{\lambda^2} \cdot \text{Aperture} \right)$$

with reference to an isotropic radiator; η is the efficiency of the antenna.

Efficiency

Consider a dish antenna pointed at

¹Notes appear on page 11.

an isotropic antenna transmitting some distance away. We know that the isotropic antenna radiates uniformly in all directions, so it is a simple(!) matter of spherical geometry to calculate how much of that power should be arriving at the dish over its whole aperture. Now we measure how much power is being received from the dish at the electrical connection to the feed—never greater than that arriving at the aperture. The ratio of power received to power arriving is the aperture efficiency.

How much efficiency should we expect? For dishes, all the books say that 55% is reasonable, and 70 to 80% is possible with very good feeds. Several amateur articles have calculated gain based on 65% efficiency, but I haven't found measured data to support any of these numbers. On the other hand, KI4VE suggests that the amateur is lucky to achieve 45-50% efficiency with a small dish and a typical "coffee-can" feed.³

For horns and lenses, 50% efficiency is also cited as typical. Thus, we should expect about the same gain from any of these antennas if the aperture area is the same.

Reciprocity

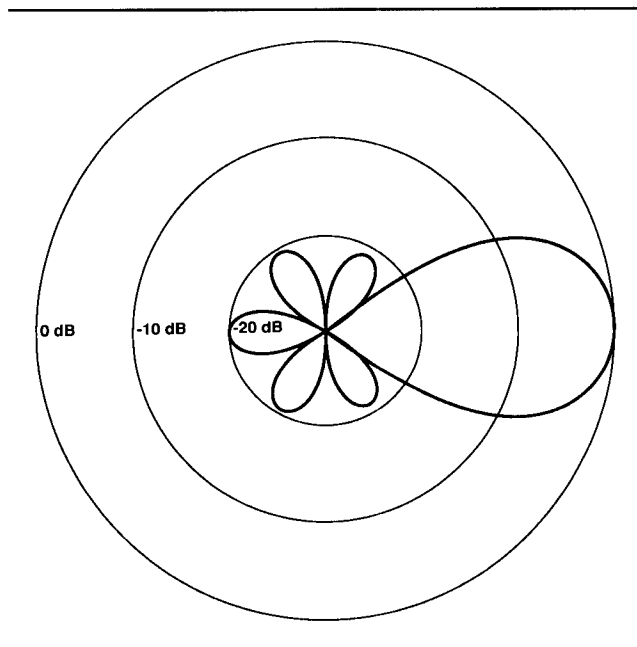
Suppose we transmit alternately with a smaller and a larger dish and note the relative power received at a distant antenna. Then if we transmit from the distant antenna and receive alternately with the same two dishes, would we expect to see the same relative power? Yes. Transmitting and receiving gains and antenna patterns are identical. This is hard to prove mathematically, but it is so.^{4, 5}

However, the relative noise received by different types of antennas may differ, even with identical antenna gains. Thus, the received signal-to-noise ratio may be better with one type of antenna than another.

Directivity and Beamwidth

Suppose an antenna has 20 dB of gain in some direction. That means it is radiating 100 times as much power in that direction as would an isotropic source, which uniformly distributes its energy over the surface of an arbitrarily large sphere that encloses the antenna. If all the energy from the 20-dB-gain antenna were beamed from the center of that same sphere, it would pass through an area 100 times smaller than the total surface of the sphere. Since there are 41,253 solid degrees in a sphere, the radiation

Fig 1—A typical antenna pattern showing the main lobe and sidelobes.



must be concentrated in 1/100th of that, or roughly 20° of beamwidth. The larger the gain, the smaller the beamwidth.

The directivity of an antenna is the maximum gain of the antenna compared to its gain averaged in all directions. It is calculated by calculating the gain, using the previous formula, with 100% efficiency.

Sidelobes

No antenna is able to radiate all the energy in one preferred direction. Some is inevitably radiated in other directions. Often there are small peaks and valleys in the radiated energy as we look in different directions (Fig 1). The peaks are referred to as sidelobes, commonly specified in *dB down from the main lobe*, or preferred direction.

Are sidelobes important? Let's suppose that we could make an antenna with a 1-degree beamwidth, and in all other directions the average radiation was 40 dB down from the main lobe. This seems like a pretty good antenna! Yet when we do the calculation, only 19.5% of the energy is in the main lobe, with the rest in the other 41252/41253 of a sphere. The maximum efficiency this antenna can have is 19.5%.

E-plane and H-plane

An antenna is a transducer which converts voltage and current on a transmission line into an electromagnetic field in space, consisting of an electric field and a magnetic field

oriented at right angles to one another. An ordinary dipole creates an electric-field pattern with a larger amplitude in planes which include the dipole than in other planes. The electric field travels in the E-plane; the H-plane, perpendicular to it, is the field in which the magnetic field travels. When we refer to polarization of an antenna, we are referring to the E-plane. However, for three-dimensional antennas like horns, dishes and lenses, it is important to consider both the E-plane and the H-plane, in order to fully use the antenna and achieve maximum gain.

Phase Center

The antenna pattern in Fig 1, and most other illustrations of antenna patterns, shows only amplitude, or average power. This is all we need to consider for most applications, but for antennas which are like optical systems, like lenses and dishes, we must also be concerned with phase, the variation in the signal as a function of time. RF and microwave signals are ac, alternating current, with voltage and current that vary sinusoidally (like waves) with time. Fig 2A shows several sine waves, all at the same frequency, the rate at which they vary with time.

Let's think about a simple example: a child's swing. We've all both ridden and pushed one at some time. If we push the swing just as it starts to move away from us, it swings higher each time. If we add a second pusher at the

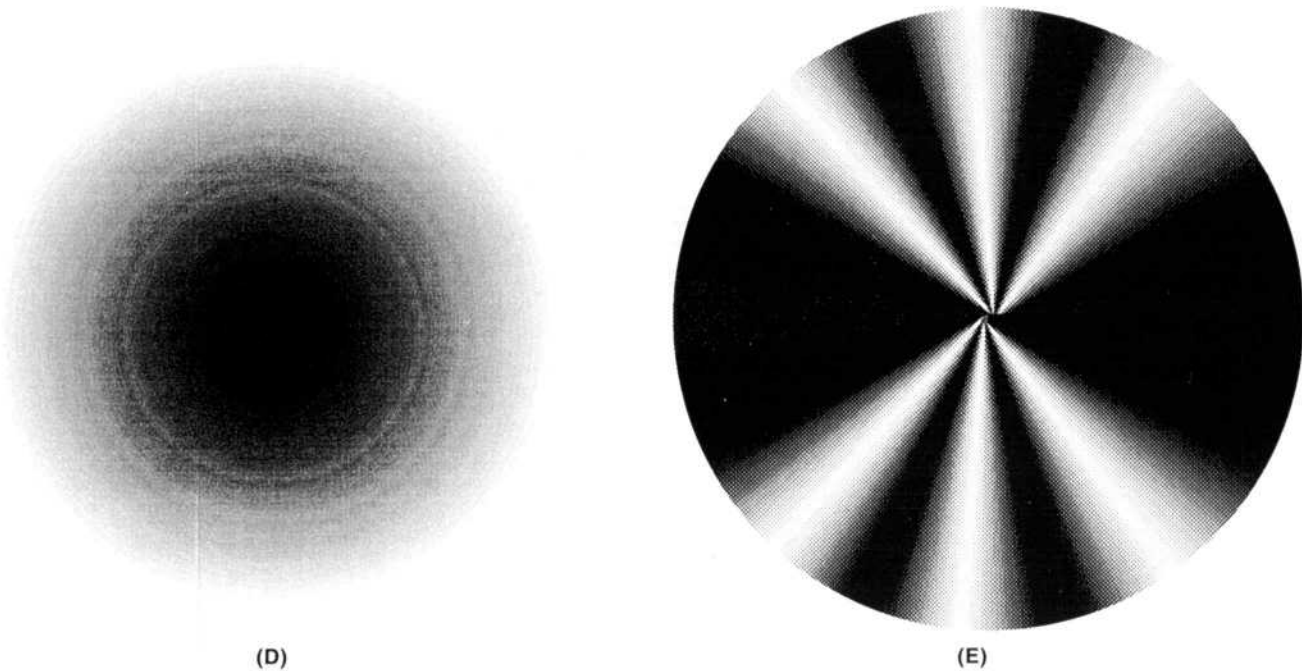
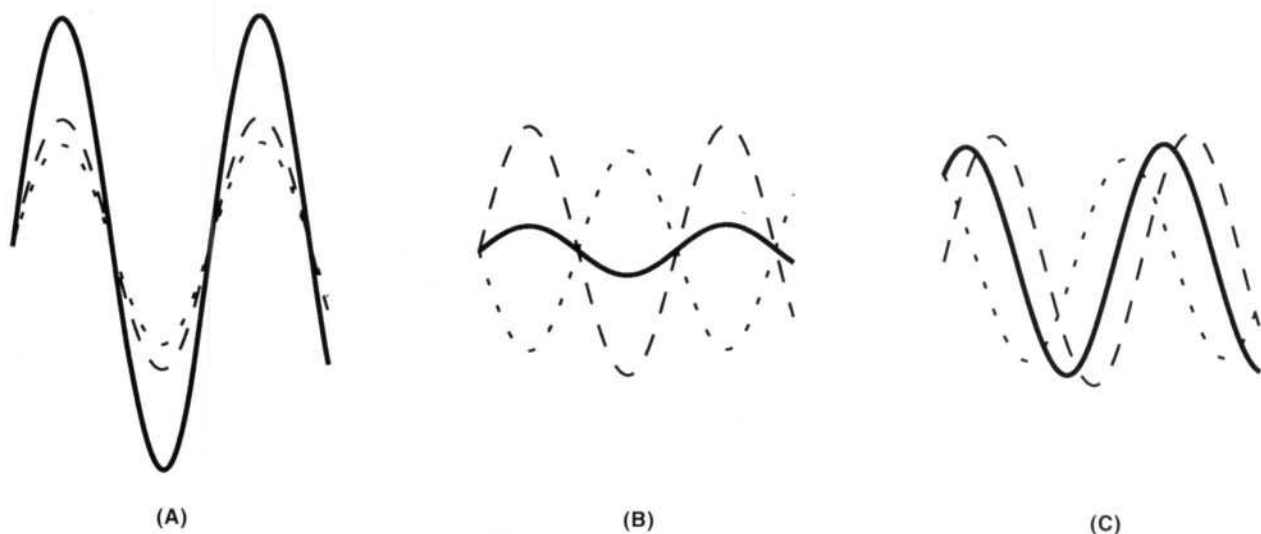


Fig 2—The result of multiple signal sources depends on the phase difference between them. At (A), two signals are shown in phase and add together. At (B), the signals are 180° out of phase and tend to cancel, while the signals at (C) are out of phase by less than 180° , with the result being a signal at a phase and amplitude different from either of the two source signals. The plot at (D) shows the amplitude around a single-source antenna, while (E) shows the interference pattern created by having two sources.

other end, it will increase faster. Now if we tie a rope to the swing seat and each pusher takes an end, we can try to add energy to the swing throughout its cycle. This will work as long as we keep the pulling synchronized with the motion of the swing, but if we get *out of phase*, we will drag it down rather than sending it higher.

The motion of a swing is periodic, and the height of the swing varies with time in a pattern similar to a sine wave

of voltage or current. Look at a sine wave in Fig 2A, considering the highest point of the waveform the height the swing travels forward, and the lowest point as the height the swing travels backward, both repeating with time. If there are two swings side-by-side and both swings arrive at their peak at the same time, they are in phase, as in Fig 2A.

When two electromagnetic waves arrive at a point in space and impinge

on an antenna, their relative phase is combined to create a voltage. If they have the same phase, their voltages add together; in Fig 2A, the two dashed waveforms are in phase and add together to form the solid waveform. On the other hand, when signals are exactly out of phase, the addition of positive voltage to negative voltage leaves only the difference, as shown in Fig 2B. If the two signals are partially out of phase, the resultant waveform

is found by adding the voltage of each at each point in time; one example is shown in Fig 2C. Notice that the amplitude of the resultant waveform is dependent on the phase difference between the two signals.

If our signal source is a point source, then all waves are coming from that one point in space. Each wave has a wavefront, like a wave arriving on a beach. The wavefront from the perfect point source has a spherical shape. Consider its *amplitude*. First, we place an antenna and power meter at some distance from the source and take a reading, then when we move the antenna around to other places that create exactly the same power reading, we will draw a sphere around the source. Thus, the amplitude has a uniform distribution like Fig 2D; dark areas have higher amplitude than lighter areas, and the amplitude decreases as we move away from the source according to the inverse square law described below (the shading has a few small concentric rings due to the limitations of computer graphics, but is really a continuous smooth function).

The *phase* of this wavefront as it propagates in space appears to also have a spherical shape. If frozen in time, one sphere would represent a positive peak of a sine wave. One half wavelength inside would be another sphere representing a negative peak of the sine wave, and another half wave inside again is a positive peak. The *phase center* of an antenna is the apparent place from which the signal emanates based on the center of a sphere of constant phase.

However, no real antenna is small enough to be a point source, so the radiation must appear to emanate from a larger area. If we consider a simple case, where the radiation appears to come from two points, then two signals will arrive at each point in space. A point in space is typically farther from one radiating point than from the other, and since the time it takes for each signal to arrive depends on the distance to each of the radiating points, there will be a phase difference between the two signals. This phase difference will be different at each point in space, depending on the relative distances, and the amplitude of the resultant signal at each point depends on the phase difference. An example of a pattern created by two radiating sources is shown in Fig 2E, where the dark areas have the greatest amplitude, due to the two signals

arriving in phase, and the light areas are areas where phase cancellation, like that of Fig 2B, has reduced the amplitude.

A well designed feed for a dish or

lens has a single phase center, so the radiation appears to emanate from a single point source. This must be so for at least the main beam, the part of the pattern that illuminates the dish or

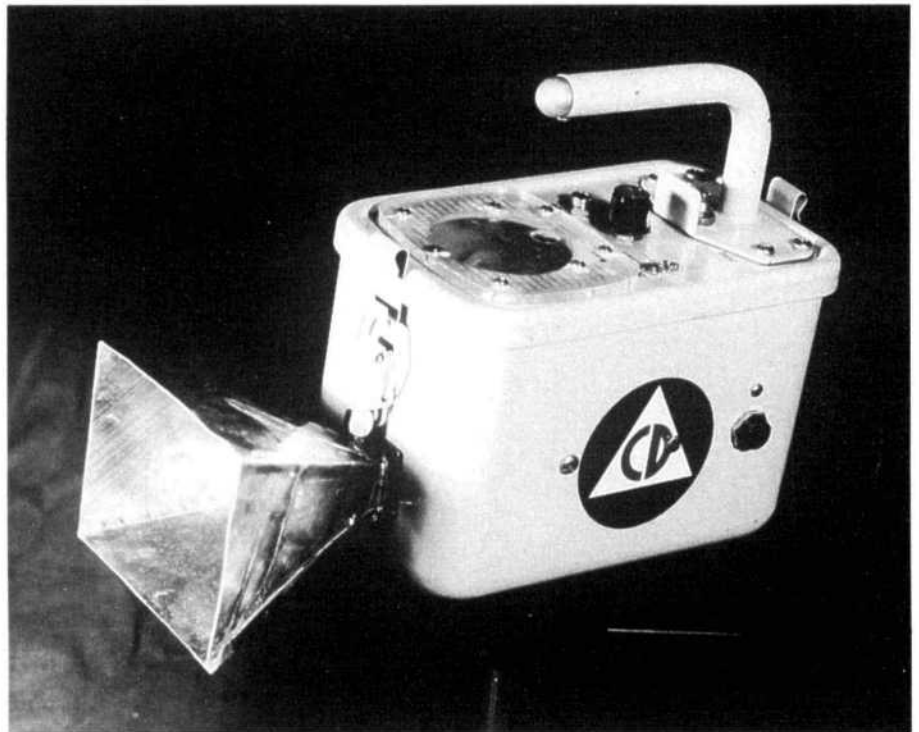


Fig 3—A homebrew horn for 10 GHz, made from flashing copper and designed using the HDL_ANT program.

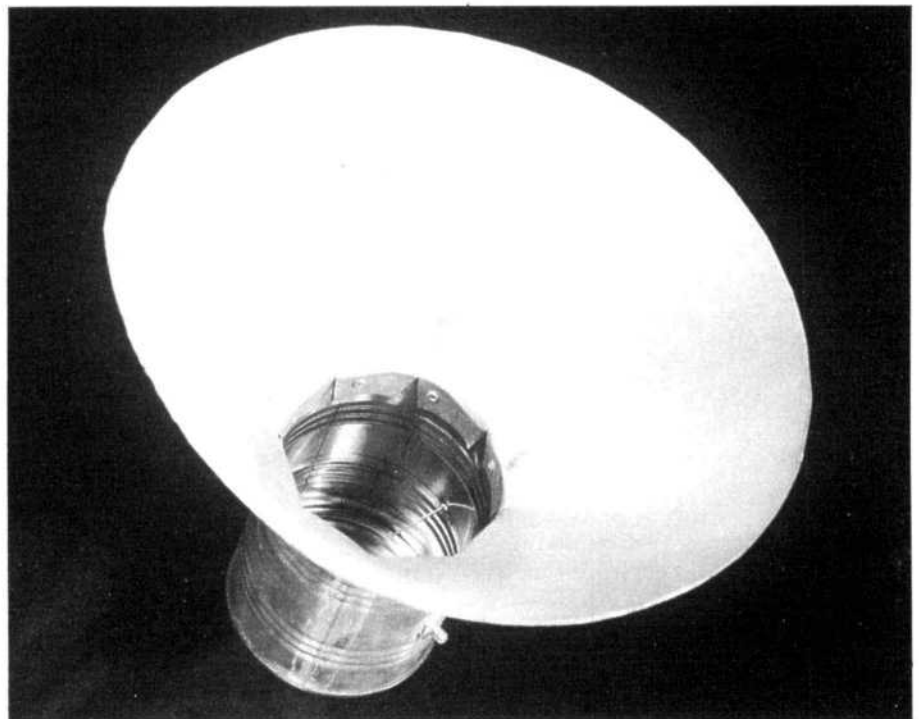


Fig 4—A homebrew conical horn for 2304 MHz.

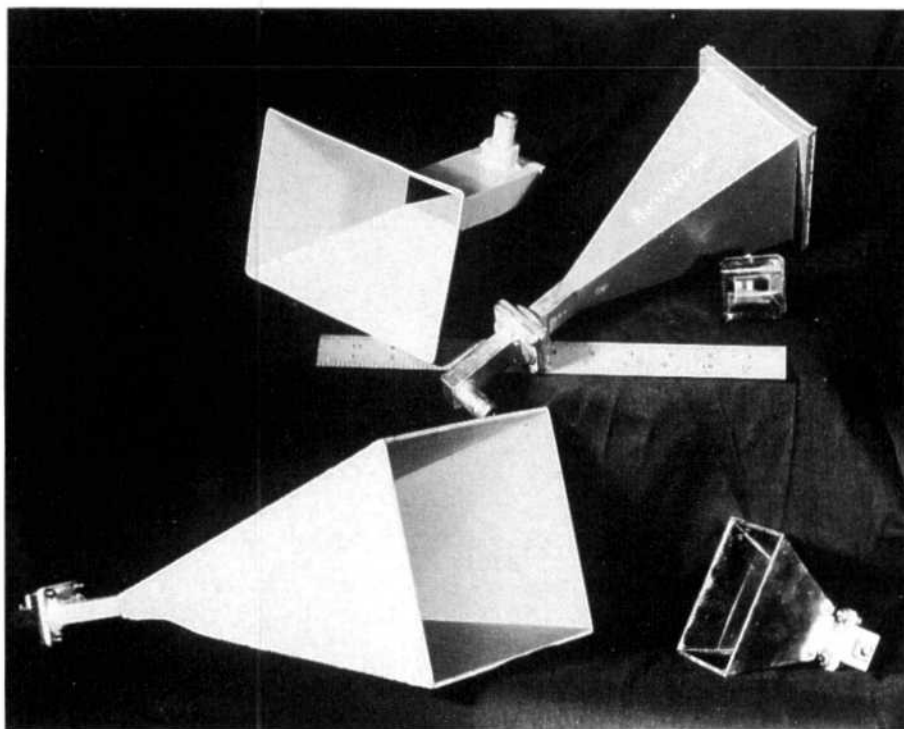


Fig 5—A variety of rectangular horn antennas.

lens. Away from this main beam, the phase center may move around and appear as multiple points, due to stray reflections and surface currents affecting the radiation pattern. However, since these other directions do not illuminate the dish or lens, they can be ignored.

Inverse Square Law

As two antennas are moved farther apart, received power decreases in proportion to the square of the distance between them; when the distance is doubled, only 1/4 as much power is received, a reduction of 6 dB. This is because the area illuminated by a given beamwidth angle increases as the square of the distance from the source, so the power per unit area must decrease by the same ratio, the square of the distance. Since the area of the receiving antenna has not changed, the received power must decrease proportionally.

The phase center pattern in Fig 2E does *not* include the effect of inverse square law in the pattern, in order to emphasize the phase cancellation. The effect of including inverse square law would be to lighten the pattern as distance from the phase center increased.

Free Lunch

Since gain is proportional to aper-

ture, larger antennas have more gain than smaller antennas, and poor efficiency can only make a small antenna worse. In spite of various dubious claims by antenna designers and manufacturers, "There's no such thing as a free lunch."⁶ All else being equal, the larger the antenna, the greater the gain. But a large antenna with poor efficiency is a waste of metal and money.

Recommended Reading

For those interested in pursuing a deeper understanding of antennas, a number of books are available. A good starting point is *The ARRL Antenna Book* and *The ARRL UHF/Microwave Experimenter's Manual*. Then there are the classic antenna books, by Kraus, Silver and Jasik.^{2,4,7} Lo and Lee have edited a more recent antenna handbook, and Love has compiled most of the significant papers on horns and dishes.^{5,8,9} For those interested in computer programming for antenna design, Sletten provides a number of routines.¹⁰ Be warned that the math gets pretty dense once you get beyond the ARRL books.

Summary

This concludes our quick tour through basic antenna concepts and definitions. Now let's apply these con-

cepts to understanding actual microwave antennas, starting with horns.

The HDL_ANT Computer Program

The intent of the *HDL_ANT* program is to aid the design of microwave antennas, not to be a whizzy graphics program. The program does the necessary calculations needed to implement a horn, dish or lens antenna, or to design an antenna range and correct the gain measurements. The basic data is entered interactively and results are presented in tabular form. If you like the results, a table of data or a template may be saved to a file for printing or further processing; if not, try another run with new data.

The C++ source code is also included, for those who wish to enhance it or simply to examine the more complex calculations not shown in the text. It has been compiled with Borland C++ version 3.1 and is available from the ARRL BBS at 203-666-0578, or can be downloaded via the Internet from ftp.cs.buffalo.edu in the /pub/ham-radio directory.

Electromagnetic Horn Antennas

A horn antenna is the ideal choice for a contest rover station. It offers moderate gain in a small, rugged package with no adjustments needed, and has a wide enough beam to be easily pointed under adverse conditions. Fig 3 is a photograph of a homebrew horn mounted on an old Geiger counter case which houses the rest of a 10-GHz wide-band FM transceiver. I have worked six grid squares on 10 GHz from Mt. Wachusett in Massachusetts using a small horn with 17.5 dB of gain.

Horn Design

An antenna may be considered as a transformer from the impedance of a transmission line to the impedance of free space, 377 ohms. A common microwave transmission line is *waveguide*, a hollow pipe carrying an electromagnetic wave.¹¹ If one dimension of the pipe is greater than a half wavelength, then the wave can propagate through the waveguide with extremely low loss. And if the end of a waveguide is simply left open, the wave will radiate out from the open end.

Practical waveguides have the larger dimension greater than a half wavelength, to allow wave propagation, but smaller than a wavelength, to suppress higher-order *modes* which

**E-plane
N1BWT 1994**



Fig 6—This full-scale template can be used to construct a horn antenna for 5760 MHz. Tape a copy to a piece of flashing copper and cut along the solid lines. Then fold at the dotted lines to form the rectangular horn. Solder the small flap to complete the horn, then solder the narrow end of the horn to a piece of waveguide. This antenna gives 13.8 dBi of gain.

can interfere with low-loss transmission. Thus the aperture of an open-ended waveguide is less than a wavelength, which does not provide much gain.

For more gain, a larger aperture is desirable, but a larger waveguide is

not. However, if the waveguide size is slowly expanded, or tapered, into a larger aperture, then more gain is achieved while preventing undesired modes from reaching the waveguide. This taper is like a funnel, called a conical horn, in cylindrical waveguide.

The conical horn for 2304 MHz shown in Fig 4 was made by pop-riveting aluminum flashing to a coffee can. With common rectangular waveguide, the taper creates a familiar pyramidal horn, like those shown in the photograph, Fig 5.

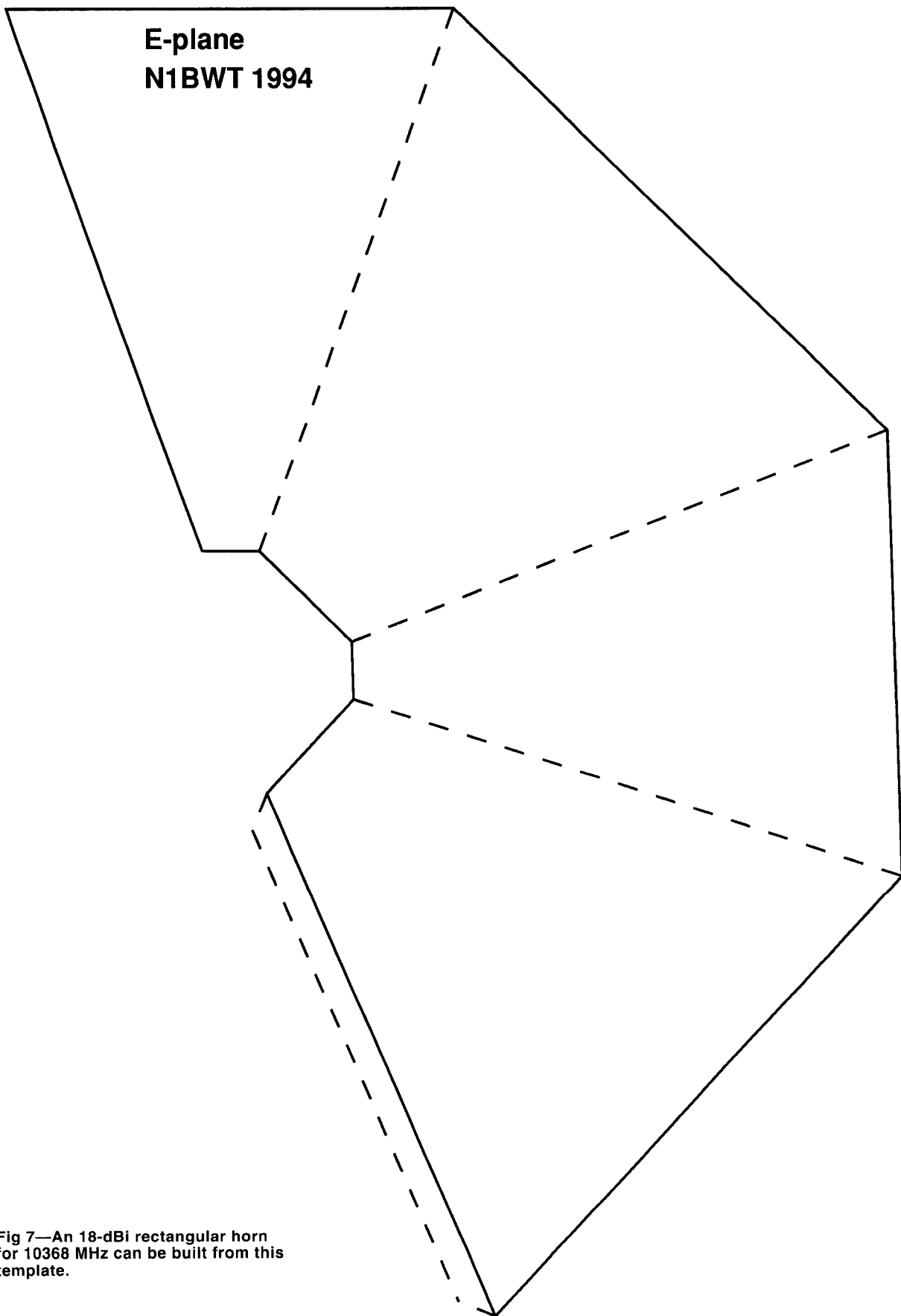


Fig 7—An 18-dBi rectangular horn for 10368 MHz can be built from this template.

Printing Postscript Files

The easiest way to print Postscript files is with a Postscript compatible laser printer. These have become more affordable and are becoming more common; for instance, the public library in my small town has one attached to a public-access computer. However, they are still roughly twice as expensive as the dot-matrix printers that most of us use with our personal computers.

An alternative to a laser printer is software that interprets Postscript language commands for display on a computer VGA display or a dot-matrix printer. I know of several versions of this type of software. Three commercial products, *GoScript*, *Ultrascript* and *Freedom of Press* perform this function. *Ghostsript*, a freeware program from the Free Software Foundation is available on many bulletin boards and Internet locations. The files are in ZIP format, so they must be downloaded, unZIPped, and installed according to the README documentation.

I have only used *Ghostsript*, version 2.5. While 286, 386 and Windows versions are available, the 286 version seems to work most reliably, even on a 386 or 486 PC. It uses Unix-style command strings which are difficult to remember, so I've included two BAT files to help: GS_VIEW.BAT for viewing on a screen, and GS_PRINT.BAT for printing on an Epson dot-matrix printer. For other brands of printer, the command will have to be changed appropriately, which will require reading of the documentation. Type GS_VIEW <filename.ps> or GS_PRINT <filename.ps> to use them. Be sure to type QUIT when you are through or your PC

may be left in an unhappy state requiring rebooting.

I've included with *HDL_ANT* a sample Postscript file, SQUARE.PS, which draws a four-inch square. Use this to make sure that templates will be drawn to scale. A sample horn template, HORN18.PS, is included, too, to get you started. If the dimensions of the printed square are slightly off, you can correct the scaling. Each template has a line near the beginning of the file:

```
1.0 1.0 scale
```

The first number is the scale factor in the x (horizontal) direction, and the second is the scale factor in the y (vertical) direction. Edit the SQUARE.PS file with an editor to change these numbers slightly; when you find a combination that prints a square exactly four inches on a side, then you have compensated for your printer. Edit these same numbers into any template to be printed on that printer and the dimensions will come out right.

I have not used any of the commercial products, but I would expect a commercial product to be much easier to install and use than freeware or shareware.

Batch Files

```
GS_VIEW.BAT
gs %1
```

```
GS_PRINT.BAT
gs -sDEVICE=epson -r60x60 %1
```

To achieve maximum gain for a given aperture size and maximum efficiency, the taper must be long enough so that the phase of the wave is nearly constant across the aperture. An optimum horn is the shortest one that approaches maximum gain; several definitions are available. The *HDL_ANT* program uses approximate dimensions from a set of tables by Cozzens to design pyramidal horn antennas with gains from 10 to 25 dB.¹² Higher gains are possible, but the length of the horn increases much faster than the gain, so very high gain horns tend to be unwieldy.

Kraus gives the following approximations for beam width in degrees:

$$W_{E\text{-plane}} = \frac{56}{A_{E\lambda}}$$

$$W_{H\text{-plane}} = \frac{67}{A_{H\lambda}}$$

and dB gain over a dipole:

$$\text{Gain} \approx 10 \log_{10}(4.5 \cdot A_{E\lambda} \cdot A_{H\lambda})$$

where $A_{E\lambda}$ is the aperture dimension in wavelengths in the E-plane and $A_{H\lambda}$ is the aperture in wavelengths dimension in the H-plane. The *HDL_ANT* program uses a more accurate gain

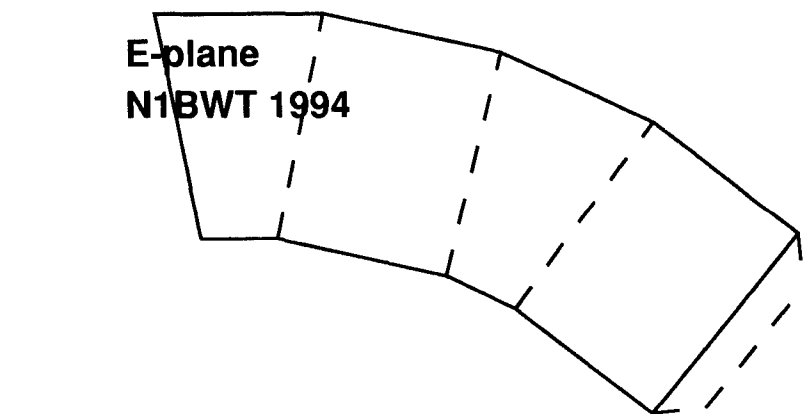


Fig 8—A template for a 10368-MHz, 8-dBi horn, suitable for feeding an f/D=0.5 dish.

algorithm which corrects the phase error of different taper lengths; for a given aperture, efficiency and gain decrease as the taper is shortened.¹³

Horn Construction

If you are fortunate enough to find a suitable surplus horn, this section is

unnecessary. Otherwise, you may want to homebrew one. Horn fabrication is quite simple, so you can homebrew them as needed, for primary antennas with moderate gain or as feeds for higher gain dishes and lenses. Performance of the finished horn almost always matches predic-

tions, with no tuning adjustments required.

The *HDL_ANT* program will design a horn with any desired gain or physical dimensions and then make a template for the horn. The template is a Postscript file; print the file on a computer printer to generate a paper template, tape the paper template to a sheet of copper or brass, cut it out, fold on the dotted lines, and solder the metal horn together on the end of a waveguide. The horn shown in Fig 3 used flashing copper from the local lumberyard, which I soldered together on the kitchen stove.

Fig 6 is a template for a nominal 14-dB horn for 5760 MHz generated by *HDL_ANT*. Try it: copy it on a copier and fold up the copy to see how easy it is to make a horn. It's almost as easy with thin copper. Fig 7 is another template example, a nominal 18-dB horn for 10368 MHz. For horns too large to fit the entire template on one sheet of paper, *HDL_ANT* prints each side on a separate sheet.

Feed Horns

For horns intended as feed horns for dishes and lenses, beam angle and phase center are more important than

horn gain. The *HDL_ANT* program calculates these values in both the E-plane and the H-plane, then allows you to enter new horn dimensions to adjust the beam angle or phase center before making a template. The phase center calculation is a difficult one involving Fresnel sines and cosines, so interactive adjustment of horn dimensions is a lot easier than having the computer try to find the right dimensions.^{14,15} The template in Fig 8 is one example of a feed horn—it may be used to make a rectangular horn optimized to feed a dish with $f/D = 0.5$ at 10 GHz.¹⁶ Feed horn design for dishes and lenses will be described in more detail in Parts 2 and 3 of this series.

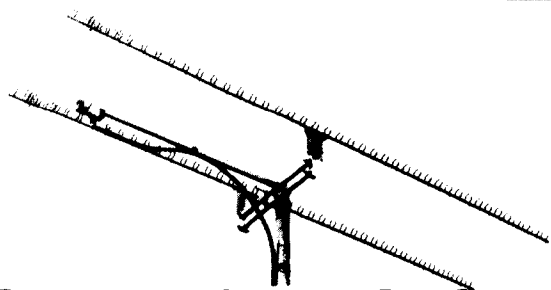
Conclusion

Horns are versatile microwave antennas, easy to design and build with predictable performance. They should be the antenna of choice for all but the highest gain applications.

Notes

- ¹ Wade, P., N1BWT, and Reilly, M., KB1VC, "Metal Lens Antennas for 10 GHz," *Proceedings of the 18th Eastern VHF/UHF Conference*, ARRL, 1992, pp 71-78.
- ² Kraus, John (W8JK), *Antennas*, McGraw Hill, 1956.

- ³ Ralston, M., K14VE, "Design Considerations for Amateur Microwave Antennas," *Proceedings of Microwave Update '88*, ARRL, 1988, pp 57-59.
- ⁴ Silver, Samuel, *Microwave Antenna Theory and Design*, McGraw-Hill, 1949. (Volume 12 of Radiation Laboratory Series, reprinted 1984.)
- ⁵ Lo, Y. T. and Lee, S. W., editors, *Antenna Handbook: theory, applications, and design*, Van Nostrand Reinhold, 1988.
- ⁶ Attributed to economist Milton Friedman.
- ⁷ Jasik, Henry and Johnson, Richard C., *Antenna Engineering Handbook*, McGraw-Hill, 1984. (Also first edition, 1961.)
- ⁸ Love, A. W., *Electromagnetic Horn Antennas*, IEEE Press, 1976.
- ⁹ Love, A. W., *Reflector Antennas*, IEEE Press, 1978.
- ¹⁰ Sletten, Carlyle J. (W1YLV), *Reflector and Lens Antennas*, Artech House, 1988.
- ¹¹ *The ARRL UHF/Microwave Experimenter's Manual*, ARRL, 1990, pp 5-21 to 5-32.
- ¹² Cozzens, D. E., "Tables Ease Horn Design," *Microwaves*, March 1966, pp 37-39.
- ¹³ Balanis, C. A., "Horn Antennas," in *Antenna Handbook: theory, applications, and design* (see Note 5).
- ¹⁴ Muehldorf, Eugen I., *The Phase Center of Horn Antennas*, reprinted in *Electromagnetic Horn Antennas* (see Note 8).
- ¹⁵ Abramowitz, Milton and Stegun, Irene A., *Handbook of Mathematical Functions*, Dover, 1972.
- ¹⁶ Evans, D., G3RPE, "Pyramidal horn feeds for paraboloidal dishes," *Radio Communication*, March 1975.



DOWN EAST MICROWAVE

Amateur Microwave Antennas and Equipment

902, 1269, 1296, 2304, 2320, 2400, 3456 MHz

TROPO, EME, WEAK SIGNAL, OSCAR MODE L, MODES, ATV, REPEATERS



LOOP YAGIS, POWER DIVIDERS, COMPLETE ARRAYS
KIT FORM OR ASSEMBLED AND TESTED

SOLID STATE LINEAR AMPLIFIERS FOR 902 & 1296 MHz

Write for Free Catalog to:

DOWN EAST MICROWAVE


Bill Olson W3HQT, Box 2310 RR1
Troy, ME 04987 (207) 948-3741

Surface Mount Chip Component Prototyping Kits—

Only \$49⁹⁵

INDIVIDUAL VALUES AVAILABLE



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

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

COMMUNICATIONS SPECIALISTS, INC.

426 West Taft Ave. • Orange, CA 92665-4296
Local (714) 998-3021 • FAX (714) 974-3420

Entire USA 1-800-854-0547

WAVELET COMPRESSION FOR IMAGE TRANSMISSION THROUGH BANDLIMITED CHANNELS

A. Langi, VE4ARM, and W. Kinsner, VE4WK

Department of Electrical and Computer Engineering,
and Telecommunications Research Laboratories
University of Manitoba
Winnipeg, Manitoba, Canada R3T 5V6
Tel.: (204) 474-6992; Fax: (204) 275-0261
eMail: kinsner@ee.umanitoba.ca

ABSTRACT

This paper studies an image compression scheme using wavelets for image transmission through bandlimited channels. During encoding, the scheme first transforms the image to a wavelet domain and then compresses the wavelet representation into an image code. Conversely, during decoding, the scheme first decompresses the image code into wavelet representation and then transforms it back to the original domain. The wavelet transform is explained from a perspective of signal representation in $L^2(R)$ space. The transform is further computed using a pyramidal algorithm. The compression is possible because (i) the wavelet representation has many small values that can be coded using fewer bits, and (ii) the wavelet basis functions are localized in space and frequency domains such that an error in the wavelet representation only locally affects the image in those domains. An experiment has been performed to compress two 256×256 greyscale images on such a scheme through (i) a transformation using a simple wavelet called DAUB4, (ii) redundancy removal by truncation the small-valued wavelet representation, and (iii) a ZIP compression (based on Shannon-Fano and Lempel-Ziv-Welch techniques). The results show that highly truncated wavelet representation ($\geq 90\%$) still provides good image quality (PSNR ≥ 30 dB) at less than 2 bits per pixel (bpp). Severe truncation still preserves general features of the image.

1. INTRODUCTION

Image compression is a problem of considerable importance because it leads to efficient usage of channel bandwidth and storage during image transmission and storage, respectively. There are many applications that require image transmission, such as high-definition television (HDTV), videophone, video conferencing, interactive slide show, facsimile, and multimedia [AnRA91], [Kins91], [Prag92], [JaJS93]. However, images require many data bits, making it impossible for real-time transmission through bandlimited channels, such as 48 kbit/s

and 112 kbit/s integrated services digital network (ISDN), as well as 9.6 kbit/s voice-grade telephone or radio channels. Non real-time transmission on such channels also takes a long time. For example, a 256×256 pixel greyscale still image with 8 bits per pixel (bpp) requires 65,536 bytes. Transmission of such an image over the voice-grade line would require at least 54.61 s. Furthermore, one HDTV format needs 60 frames of 1280×720 pixels per second. Using 24 bpp color pixels, this HDTV format would require a channel capacity of 1,440 Mbits/s.

Image compression can improve transmission performance by reducing the number of bits that must be transmitted through the channel. As shown in Fig. 1, an image compressor compactly represents the image prior to channel encoding. At the receiving end, the demodulation and signal decoding obtain the compressed image, and an image decompressor converts the compressed image back to a viewable image. The transmission now requires a shorter time because the number of bits that must be transmitted has been reduced. Image compression also reduces the storage space requirement for image storage.

Although there exists several image compression methods, we still need better ones because current methods are still inadequate, especially for image transmis-

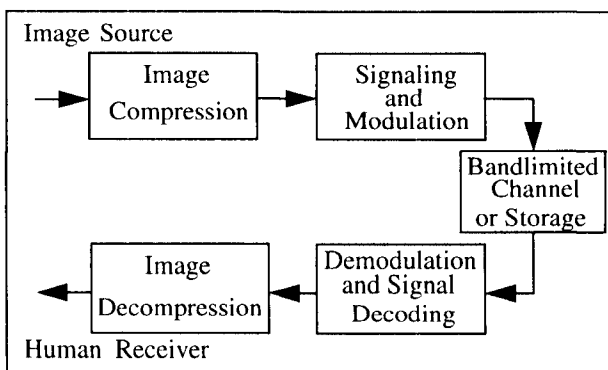


Fig. 1. Efficient image transmission system.

sion through bandlimited channels. For example, a sophisticated joint photograph expert group (JPEG) method needs 1.6 bpp for 24 bpp color images. This means that HDTV with 30 frames per second would still require a 4.423 Mbit/s channel capacity. Low bit rate compression methods have been relying on *lossy* type of compressing data, in which reducing the bit rate has caused a removal of some image information. Thus, there is a limit on how far a method can reduce the bit rate, while keeping information distortion sufficiently low.

The performance measurement of a compression technique is based on how successful it is to reduce the bit rate while maintaining image quality [Kins91]. The other performance aspects are algorithm complexity and communication delay [JaJS93]. This paper mainly focuses on the interplay between the first two aspects: reducing the bit-rate and maintaining quality.

The wavelet technique (or subband coding in general) has been emerging as an important class of image compression, in that it has an efficient representation with few visible distortions [Kins91], [Mall89], [JaJS93]. It is more promising than the JPEG standard, due to (i) a match between wavelet processing and the underlying structure of visual perception model, and (ii) a match between characters of wavelet analysis and natural images, i.e., higher frequency tends to have shorter spatial support [JaJS93]. They are also attractive because wavelet techniques are quite fast and easy to implement. Furthermore, wavelet techniques can provide special representation such as multi-resolution encoding [Mall89].

This paper shows the potential of using wavelet representation for image compression. The wavelet representation is introduced through a signal representation theory, and computed through a wavelet transform pair in Section 2. Section 3 shows a wavelet compression scheme based on the wavelet transform pair to convert spatial-domain images to and from the wavelet domain. The scheme then compresses the image in wavelet domain using a combination of lossless (no information loss) or lossy (some minor losses) techniques of data compression. Preliminary experiments on two 256×256 greyscale images (*lena.img* and *camera.img*) using a simple wavelet (DAUB4) and a simple compression (truncation and ZIP) show the potential of the scheme. Those images are used because they are standard and easily obtained by other researchers. At 4 bpp, the distortion is almost unnoticeable. At 2 bpp, the distortion is noticeable, but the errors are of salt and pepper noise type such that a simple filtering may be able to eliminate them. At highly compressed

images, the details are lost but the general features are still preserved.

2. WAVELET TRANSFORM OF SIGNALS

The purpose of this section is to construct a *discrete wavelet transform* (DWT) for discrete signal representation. This construction is explained through a theory of signal representation in a special space, called $L^2(R)$, which is a space for all signals with bounded energy (square integrable) [Mall89]. This restriction does not really affect the application generality because most of the signals that are of interest to us are in that space. Although we use a one-dimensional signal as an example, we can extend the results to two-dimensional signals such as images.

2.1. Signal Representation in $L^2(R)$

In $L^2(R)$, a signal $x(t)$ can be represented in various domains using various sets of basis functions. Let a domain be spanned by basis functions $\varphi_i(t)$, having reciprocal signals $\theta_i(t)$, where $i \in Z$ (integer numbers). The representation of $x(t)$ in this domain is a set of scalars α_i [Fran69], satisfying

$$\alpha_i = \langle \mathbf{x}, \theta_i \rangle \quad (1)$$

where $\langle \bullet, \bullet \rangle$ is an inner product, defined in $L^2(R)$ as

$$\langle \mathbf{x}, \theta_i \rangle = \int_R x(t) \theta_i(t) dt \quad (2)$$

Equation (2) is often called the cross correlation between signals $x(t)$ and $\theta_i(t)$. The $x(t)$ can be reconstructed back through

$$x(t) = \sum_i \alpha_i \varphi_i(t) \quad (3)$$

Thus $x(t)$ is a linear combination of $\varphi_i(t)$.

It may happen that the basis are not countable. In other word, we have $\varphi(\tau, t)$ and $\theta(\tau, t)$, called *basis kernel*, instead of $\varphi_i(t)$ and $\theta_i(t)$. In such a continuous case, the new representation of $x(t)$, which is called $u(\tau)$, becomes

$$u(\tau) = \int_R x(t) \theta(\tau, t) dt \quad t, \tau \in R \quad (4)$$

and the reconstruction becomes

$$x(t) = \int_R u(\tau) \varphi(t, \tau) d\tau \quad (5)$$

In many applications, it is desirable to have the same set for basis and the reciprocal, or equivalently $\varphi_i(t)$ is

equal to $\theta_i(t)$, because of the difficulties obtaining and dealing with certain $\theta_i(t)$. In this case, the basis functions are *self-reciprocal*, or *orthogonal*. Let the *norm* of a signal $f(t)$ be defined as

$$\|f(t)\| = \sqrt{\langle f(t), f(t) \rangle} \quad (6)$$

If the norm of each basis function is one, the basis functions are *orthonormal*.

There are many basis functions available. In fact, there is a very large number of them. However, only a few of them are useful, including wavelets.

2.2. Wavelets as Basis Functions

As basis functions, wavelets have several special properties. First, in the frequency domain, a wavelet can be seen as a special bandpass signal. A wavelet operating at a higher frequency has a wider bandwidth. The bandwidth is proportional to the centre frequency of the signal. This is useful for short-time signal analysis because wavelets can provide enough analysis resolution in both original and frequency domains. Second, the wavelets in a set of basis functions are scaled (by a factor a) and translated (by a time-shift parameter b) versions of a single wavelet prototype $\psi(t)$ [RiVe91]. In a mathematical notation, each wavelet is in the form of

$$\psi_{a,b}(t) = \frac{1}{\sqrt{a}} \psi\left(\frac{t-b}{a}\right) \quad (7)$$

where a is the scaling factor and b is the translation parameter. Thus, the wavelets have self similarity, and useful in revealing self-similarity aspect of signals. This also leads to fast algorithms. Third, in contrast with Fourier representation, there are more than one wavelet prototypes. In Fourier representation, the prototype is $e^{j\omega t}$ only. Thus the same wavelet tools can be applied to many different sets of wavelet basis.

The selection of a and b leads to a different class of wavelet representation. In general, the $\psi_{a,b}$ are not orthogonal because a and b can be any real, continuous value. In this case, applying Eq. (2) through (6) is more difficult because we must find the reciprocal basis for each wavelet set. However with some restrictions, we can use those equations as if the wavelets were orthogonal basis functions. The restrictions are that the wavelet prototype $\psi(t)$ must be (i) finite energy and (ii) bandpass (no DC component) [RiVe91]. In this situation, if we use Eq. (5) and (6), we have a *continuous wavelet transform* (CWT) that operates on continuous signal using wavelet basis kernel. Notice that some modification is needed, because we have two parameters a and b instead of just one τ as in the Eq. (5) and Eq. (6). We then have

$$u(a,b) = \int_R x(t) \psi_{a,b}(t) dt \quad t, a, b \in R \quad (8)$$

and

$$x(t) = \int_R \left(\int_R u(a,b) \psi_{a,b}(t) db \right) da \quad t, a, b \in R \quad (9)$$

It is important to notice that by changing the parameter a and b , we can position a wavelet in any location in the time-frequency plane (phase plane). The parameter a is a scaling parameter which changes the frequency positions of the wavelet. A larger a results in a lower center frequency, smaller bandwidth, and larger time support of the wavelet. However, in a logarithmic frequency scale, different values of a result in different bandpass signals of the same bandwidth. Furthermore, parameter b is a time shift parameter. It changes the time position of the wavelet. Thus wavelet analysis reveals both time and frequency characteristics of the signal.

2.3. Wavelet Series

We may want to use discrete a and b because continuous a and b lead to redundant representations. One optimal selection is a dyadic sequence of $a = 2^j$ and $b = 2^j k$ [Mall89], resulting in orthogonal wavelets

$$\psi_{j,k}(t) = \sqrt{2^{-j}} \psi(2^{-j}t - k) \quad (10)$$

We can now operate using Eq. (2) and (4). As before, Eq. (2) and (4) must be modified to use these basis signals because they have two integer indices, j and k , instead of one i . This modification results in a wavelet series representation.

2.4. Discrete Wavelet Transform

It is natural to extend the wavelet series for representing discrete signals using countable, discrete basis, that leads to DWT. Here, a certain selection of wavelet prototype and time-scale parameters leads to orthonormal wavelets. Mallat uses multi-resolution signal decomposition [Mall89] to obtain DWT, outlined here. The previous equations are still useful with slight modifications. Since we deal with countable basis functions, we use Eq. (2) and (4) as our basic transform pair, with Eq. (10) as the source of the basis. Furthermore, for digital signals, $x(t)$ is represented by $x[n]$, where $n \in Z$ (integer numbers).

The DWT can be developed through a multiresolution signal decomposition. Let us introduce a space V_0 which is a subspace of $L^2(R)$. The signal $x(t)$ lies in this space. The signal can be decomposed into several sig-

nals, which later called wavelet decomposition of $x(t)$ (analogy to Fourier decomposition of a signal into its frequency components). Conversely, we can use the decomposed signals to reconstruct $x(t)$ without any loss. The decomposed signals exist in decomposed subspaces created in a *pyramidal* structure. To explain this space structure, we first decompose V_0 into two orthogonal complement spaces called V_1 and O_1 denoted as

$$V_1 \oplus O_1 = V_0 \quad (11)$$

Orthogonal complement means

$$\begin{aligned} V_1 \cup O_1 &= V_0 \\ V_1 \cap O_1 &= \{\mathbf{0}\} \\ \mathbf{v} \in V_1, \mathbf{o} \in O_1 &\Rightarrow \langle \mathbf{v}, \mathbf{o} \rangle = 0 \end{aligned} \quad (12)$$

where $\{\mathbf{0}\}$ is a null set. Second similar decomposition is from V_1 into V_2 and O_2 . The decomposition continues down to J th decomposition, where V_{j-1} is decomposed into V_j and O_j , where $J \in \mathbb{Z}, J > 0$. Thus, for every j , where $j \in \mathbb{Z}, J \leq j \leq 1$, and $J > 0$, we have

$$V_j \oplus O_j = V_{j-1} \quad (13)$$

Mallat shows that there exists $\psi(t)$ such that $\psi_{j,k}(t)$ as defined in Eq. (10) are basis functions of O_j . The basis functions are orthonormal in both O_j and $L^2(\mathbb{R})$ [Mall89]. Furthermore, Mallat showed that there exists $\phi(t)$ in the space $L^2(\mathbb{R})$ such that

$$\phi_{j,k}(t) = \sqrt{2^{-j}} \phi(2^{-j}t - k) \quad (14)$$

are orthonormal basis of V_j .

We are now ready to define the DWT. Let $x[n]$ be a digital signal of limited energy, i.e.,

$$\sum_n |x[n]|^2 < \infty \quad (15)$$

(This implies $x[n] \in l^2(\mathbb{Z})$). Let also continuous orthonormal signals $\{\psi_{j,k}(t)\}$ and $\{\phi_{j,k}(t)\}$

($t \in \mathbb{R}$ and $j, k \in \mathbb{Z}$) span $L^2(\mathbb{R})$. Finally, associate $x[n]$ with $f(t) \in L^2(\mathbb{R})$ according to

$$f(t) = \sum_k x[k] \phi_{0,k}(t) \quad (16)$$

The DWT of $x[n]$ is then a mapping from $l^2(\mathbb{Z})$ to $l^2(\mathbb{Z}^2)$ resulting in a set of real numbers $\{c_{j,k}, d_{j,k}\}$ (called wavelet coefficients), according to

$$c_{j,k} = \langle f(t), \psi_{j,k}(t) \rangle \equiv \int_{-\infty}^{\infty} f(t) \psi_{j,k}(t) dt \quad (17a)$$

$$d_{j,k} = \langle f(t), \phi_{j,k}(t) \rangle \equiv \int_{-\infty}^{\infty} f(t) \phi_{j,k}(t) dt \quad (17b)$$

as in Eq. (1).

In practice, we restrict $x[n]$ to be of finite length, with N elements. In this case, J is any integer between 1 and $\log_2 N$. (In this work, we set J to $(\log_2 N) - 1$, thus the description of DWT in [Pres91a] is directly applicable). Index j is called *scale*, ranging from 1, 2, ..., to J , while k is 0, 1, ..., to $(2^j N) - 1$. Although DWT can be defined for complex signals, we have limited the Eq. (2) and Eq. (3) to real input and basis signals only.

Orthonormality of signals $\{\psi_{j,k}(t)\}$ and $\{\phi_{j,k}(t)\}$ implies that the inverse DWT gives back $x[n]$ from $\{c_{j,k}, d_{j,k}\}$ through

$$\begin{aligned} f(t) = & \sum_{j=1}^J \sum_{k=0}^{2^j N - 1} c_{j,k} \psi_{j,k}(t) + \\ & \sum_{k=0}^{2^j N - 1} d_{j,k} \phi_{j,k}(t); \end{aligned} \quad (18)$$

as in Eq. (3), and then

$$x[n] = \langle f(t), \phi_{0,n}(t) \rangle \quad (19)$$

2.5. Fast Pyramidal Algorithm for DWT

Given a discrete signal $x[n]$, one can actually compute the basis inner products in Eq. (17) using a matrix multiplication, according to

$$\begin{bmatrix} c_{0,0} \\ \dots \\ c_{j,k} \\ \dots \\ d_{J,k} \end{bmatrix} = \begin{bmatrix} \psi_{0,0}[0] & \dots & \psi_{0,0}[N-1] \\ \dots & \dots & \dots \\ \psi_{j,k}[0] & \dots & \psi_{j,k}[N-1] \\ \dots & \dots & \dots \\ \phi_{J,k}[0] & \dots & \phi_{J,k}[N-1] \end{bmatrix} \begin{bmatrix} x[0] \\ \dots \\ x[N-1] \end{bmatrix} \quad (20)$$

Here, the matrix elements are the sampled version of signals $\psi_{j,k}(t)$ and $\phi_{j,k}(t)$, with each signal becomes a

row of the matrix. If the length of the samples is N , the matrix is $N \times N$, and the complexity becomes $O(N^2)$.

To reduce the DWT complexity, a *pyramidal algorithm* can be used based on the multiresolution decomposition explained before. In the space V_j , $f(t)$ can be represented by

$$d_{j,k} = \langle f(t), \phi_{j,k}(t) \rangle \equiv \int_{-\infty}^{\infty} f(t) \phi_{j,k}(t) dt \quad (21)$$

Since each $\phi_{j,k}(t)$ is a member of both V_j and V_{j-1} , while the set of $\phi_{j-1,k}(t)$ is an orthonormal basis of V_{j-1} , we can express any $\phi_{j,k}(t)$ as

$$\phi_{j,k}(t) = \sum_l \langle \phi_{j,k}(t), \phi_{j-1,l}(t) \rangle \phi_{j-1,l}(t) \quad (22)$$

By changing the integration variable in the inner-product, it is easy to show that there exists $h[n]$ defined as

$$h[n] \equiv \frac{1}{\sqrt{2}} \int_{-\infty}^{\infty} \phi\left(\frac{1}{2}t\right) \phi(t-n) dt \quad (23)$$

such that

$$\langle \phi_{j,k}(t), \phi_{j-1,l}(t) \rangle = h[l-2k] \quad (24)$$

Thus Eq. (20) becomes

$$\phi_{j,k}(t) = \sum_l h[l-2k] \phi_{j-1,l}(t) \quad (25)$$

Combining Eq. (21) and Eq. (25), and applying inner-product properties [Fran69], we obtain

$$d_{j,k} = \sum_l h[l-2k] d_{j-1,l} \quad (26)$$

This relationship is very important, because one can efficiently compute $d_{j,k}$ from the previous $d_{j-1,l}$. Thus, by defining $x[n]$ as $d_{0,n}$, one can iteratively calculate all subsequent $d_{j,k}$ for $j = 1, 2, \dots$, using Eq. (26). Observe that Eq. (26) is essentially a filtering process of $d_{j-1,l}$ using a non-recursive filter of impulse responses $h[n]$, followed by subsampling by two. If we call this operation as H . Then

$$d_{j,k} = H \cdot d_{j-1,l} \equiv \sum_l h[l-2k] d_{j-1,l} \quad (27)$$

We can arrive with a similar relation for $c_{j,k}$, since $\psi_{j,k}(t)$ is a member of both W_j and V_{j-1} , while the set of $\phi_{j-1,k}(t)$ is an orthonormal basis of V_{j-1} . Using similar derivation yields

$$g[n] \equiv \frac{1}{\sqrt{2}} \int_{-\infty}^{\infty} \psi\left(\frac{1}{2}t\right) \phi(t-n) dt \quad (28)$$

and then, after defining an operator G ,

$$c_{j,k} = G \cdot d_{j-1,l} \equiv \sum_l g[l-2k] d_{j-1,l} \quad (29)$$

Both Eqs. (27) and (29) replace Eqs. (17b) and (17a), respectively.

Thus in this pyramidal algorithm, the DWT becomes a processing of input signal $x[n]$ through a pyramid of operators H and G for $j = 1$ to J . In each stage of j , the outputs of the operator G are collected as $c_{j,k}$, while the outputs of the operator H are reapplied as inputs for the next stage.

Although the above derivation is for the forward DWT, similar approach can be used to derive 'upside-down' pyramidal algorithm for the inverse of DWT. In fact, the derivation results in a simple relation between the lower stage to its next upper stage as follows

$$d_{j-1,l} = H^* d_{j,k} + G^* c_{j,k} \quad (30)$$

where H^* and G^* are the adjoints of H and G [Daub88].

To show that this algorithm is efficient, consider a DWT of $x[n]$ having N samples. Suppose that the total of computational cost of one input sample in a stage is a constant c . At the first stage, the cost is clearly Nc . At the second stage, there are only $N/2$ samples to be processed due to the subsampling process, resulting in a cost of $Nc/2$. Similarly, the third stage requires $Nc/4$. This process can continue until there is no input available for the next stage. Thus total complexity is

$$Nc \left(1 + \frac{1}{2} + \frac{1}{4} + \dots \right) \leq 2Nc \quad (31)$$

which is proportional to Nc instead of N^2 . We can estimate the c if we know the length of $h[n]$ and $g[n]$. Suppose the lengths of $h[n]$ and $g[n]$ are the same, which is $2L$, for an $L \in \mathbb{Z}$. Then Nc must be the number of multiply-and-accumulate (MAC) processes to complete both convolutions in Eq. (17) and Eq. (19) for all input samples at the first stage, which must be equal to $2LN$. Thus, total MAC processes to complete the DWT is $4LN$, as opposed to N^2 . Table 1 shows the complexity comparison with and without pyramidal algorithm.

Table 1. Comparison of complexity of DWT and pyramidal DWT, for N input sample and filters of length $2L$.

| Algorithm | $O()$ | Complexity $N=64, L=2$ |
|---------------------|-------|---------------------------|
| Basis Inner Product | N^2 | 4096 |
| Pyramidal Algorithm | $4LN$ | 512 |

2.6. Daubechies Wavelets

Ingrid Daubechies has derived a class of compactly supported wavelets that are compatible with Mallat's multiresolution analysis and pyramidal algorithm. Forcing the wavelets to be compactly supported while maintaining the compatibility, she obtained a method of constructing such wavelets, as well as some properties [Daub88], such as

$$\sum_n h[n] = \sqrt{2}$$

$$\sum_n g[n] = 0 \quad (32)$$

$$g[n] = (-1)^n h[2m+1-n] \quad m \in \mathbb{Z}$$

The simplest Daubechies wavelet is the DAUB4, with L in Table 1 is two. The filter part of H has four impulse responses $h[0]$, $h[1]$, $h[2]$, and $h[3]$, with values

$$\begin{aligned} h[0] &= 0.48296291 \\ h[1] &= 0.8365163 \\ h[2] &= 0.22414387 \\ h[3] &= -0.12940952 \end{aligned} \quad (33)$$

The filter part of G also has impulse responses $g[0]$, $g[1]$, $g[2]$, and $g[3]$ which are related to $h[n]$ according to Eq. (32). Here, we select $m=1$, such that $h[n]$ and $g[n]$ have a similar filter structure (down to tap positions), simplifying the implementation. This option of m also results in wavelet and scaling prototypes of the same supports.

2.7. Extending to Two-Dimensional Signals

A scheme to extend the DAUB4 DWT for two-dimensional signals is similar to that of the Fourier transform. Here, the computation is in two steps [Pres91a]. First, we apply a one-dimensional DWT sequentially on the columns and replace each column of the image with its DWT results. Second, we redo the similar transformation, but now on the rows, resulting in two-dimensional wavelet representation. Since the transform is orthogo-

nal, the number of representation data is the same with the number of pixel in the original image.

3. WAVELET COMPRESSION

3.1. Basic Scheme

Lossless compression works by identifying redundant data and removing them [Kins91]. The redundancy may come from non uniform distribution of data as well as data correlation. If more compression is still necessary, the scheme further looks for irrelevant data, converts them into redundant data, and then removes them. The compression then becomes lossy because the irrelevant data cannot be recovered.

The irrelevancy can be defined according to different criteria, ranging from subjective fidelity criteria to objective fidelity criteria [GoWi87]. Irrelevant data may be those with little contribution to the fidelity. They may also be those containing little information in the Shannon information theory sense. Clearly, one can associate a measure on irrelevancy, reflected in a quality measure.

Compression methods must then concentrate on redundancy removal and irrelevancy reduction [JaJS93]. Redundancy removal methods include predictors and transforms, while irrelevancy reduction methods include quantization and data elimination. They are often combined. For example, quantization may take place in the transform domain and/or predictor error.

In wavelet approach, it is possible to employ both predictors and transforms. The wavelet transform results carry both time and frequency samples that may have sample correlation, thus predictors can be used. Furthermore, the transform itself may have removed the redundancy and made the irrelevant data more visible in the wavelet domain. We thus study some of the possibilities through experimentation. Especially, we try to find parameters that govern the irrelevancy and redundancy.

3.2. Experimental Results

In this experiment, we study the role of the wavelet coefficient *magnitude* in governing irrelevancy. We setup the experiment to study its effect on quality (related to irrelevancy) as well as bit rate (related to redundancy). Based on a scheme shown in Fig. 2., the scheme first takes the DWT of the original image. As the primary test data, we have selected *lena.img*. A less intensive experiment was performed also on *camera.img* for confirmation (see Fig. 3(a) and Fig. 4(a)). The inverse DWT can convert the image back from wavelet domain to the spatial domain for viewing. We can then perform various processes on the transformed data. By observing the processing effects on the quality

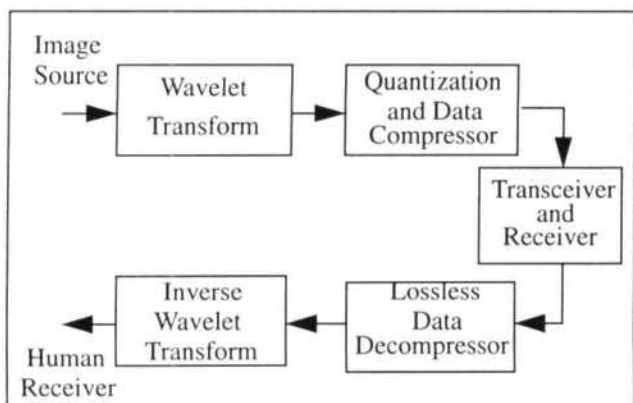


Fig. 2. A compression scheme using wavelets.

as well as the bit rate, we can design a compression scheme.

To facilitate the quality measurement, a routine computes the peak signal-to-noise ratio (PSNR) according to (in dB)

$$PSNR = 10 \log \frac{255^2}{MSE} \quad (34)$$

The mean square error (MSE) is the average of the energy of the difference between the original and the

reconstructed images. We loosely define good quality as having a PSNR of greater than 30 dB.

One way to study the magnitude effect is through a coefficient truncation scheme. Here, we use a truncation threshold, such that coefficients with absolute values smaller than the threshold are considered irrelevant, and converted into zero to become redundant data. Severe truncation leads to data losses, but resulting in an efficient compression. Thus, we study the loss in PSNR and subjective quality of the image, while at the same time observe the reduction in the size of the image code file.

Our observation reveals that the smaller the magnitude of the coefficients, the more irrelevant the coefficients are. Increasing the threshold results in more coefficients being truncated. Figures 3(b) to 3(f) and Table 2 show the effects of various degrees of truncation to the image quality. The truncation results in good quality at up to 90% truncation. As shown in Fig. 5, there are only few coefficients that are responsible for total fidelity. Those coefficients must have high coefficient values, because they survive the truncation.

One explanation is that the wavelet transform is an orthonormal transform following the principles of Eq. (1) and (3). Such a transform satisfies the Parseval's

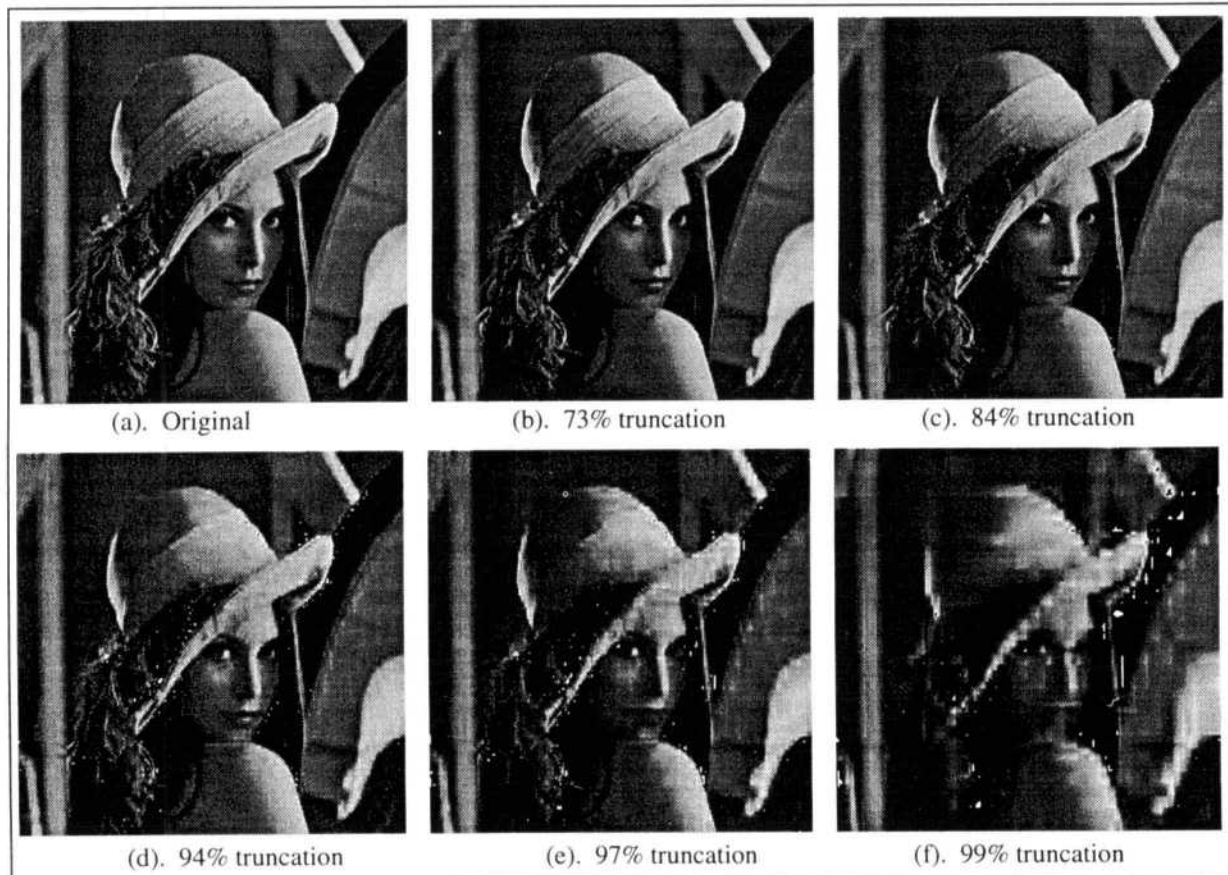


Fig. 3. Effects of various degrees of truncation on the image quality. See also Table 2.

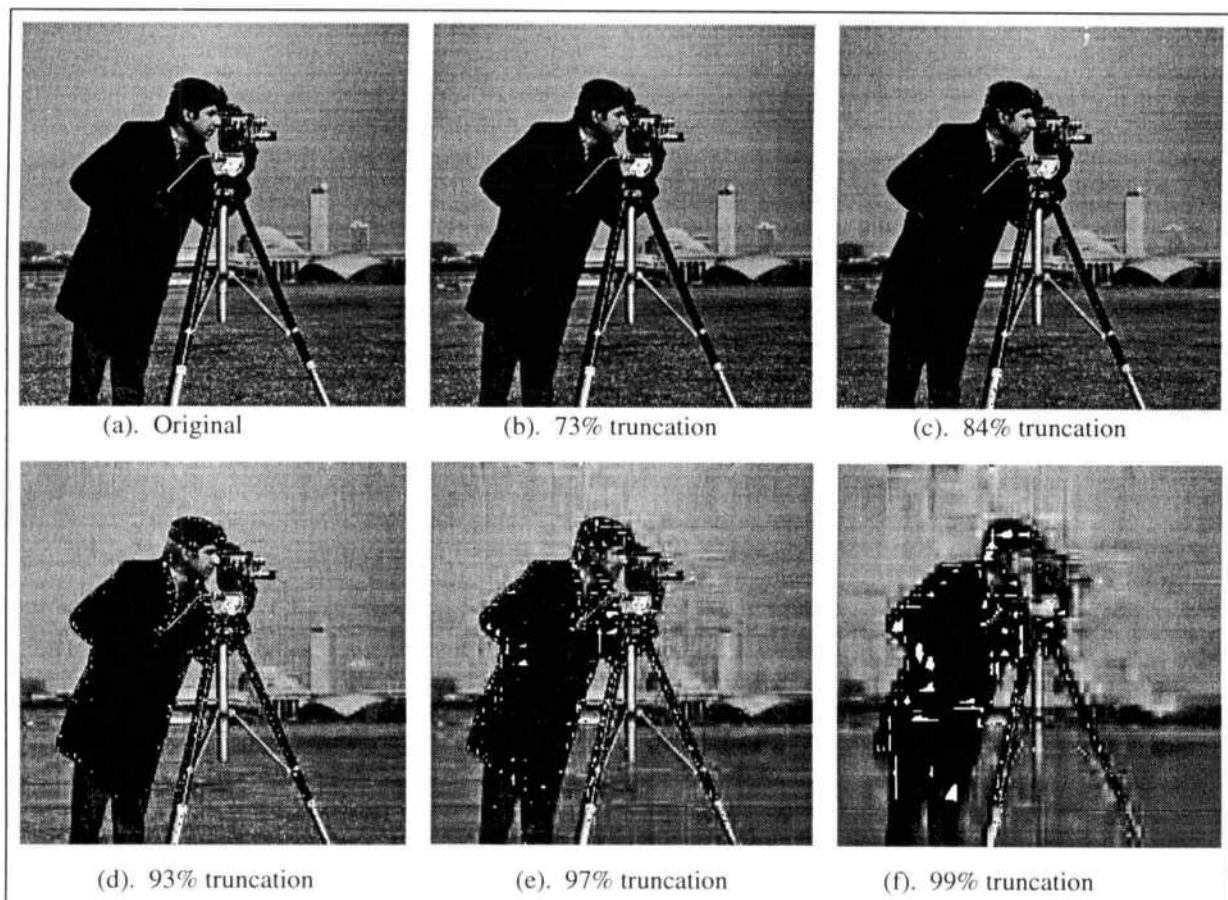


Fig. 4. Effects of various degrees of truncation on the image quality. See also Table 3.

Table 2. The PSNR values and bit rates of the images shown in Fig. 3.

| % Truncation | PSNR (dB) | Bit Rate (bpp) |
|--------------|-----------|----------------|
| 77 | 39.16 | 3.8 |
| 86 | 33.23 | 2.5 |
| 94 | 27.16 | 1.4 |
| 97 | 23.76 | 0.7 |
| 99 | 21.08 | 0.36 |

relation which equates both energies in the original and wavelet domains [Fran69]. Since the wavelet basis is orthonormal, the greater the magnitude of one coefficient, the higher its contribution to the energy. Thus truncating the such coefficients would result in high MSE values, reducing the PSNR.

For more truncation, the distortion is localized and looks like salt-and-pepper noise, as shown in Fig.3(b) to 3(e). This is due to the fact that the coefficients respon-

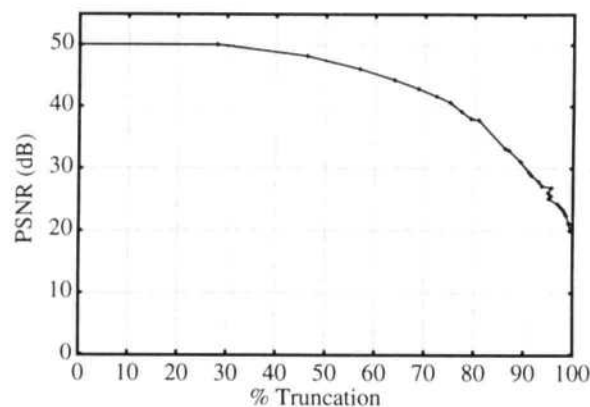


Fig. 5. Effects of % of truncation on the image quality for lena .img.

sible for the constructing the pixels have been eliminated. This also verifies one of wavelet properties of being localized in both spatial and frequency domain. Thus quantization error in the wavelet domain does not translate in distortion distributed all over the image. Low pass or median filters should be able to reduce the effects of such a distortion. Severe truncations still pre-

serve the general looks, as shown in Fig. 3(f). The images are very distorted, but the general feature is still observable.

The set of truncated coefficients has an unbalanced distribution of zeros now such that a lossless compression should be able to compress it. We can then use the ZIP compression to compress the coefficients. As shown in Fig. 6, 90% of truncation translates to 2 bpp representation. The ZIP program was able to identify data repetition in the truncated coefficients, and encoded them compactly.

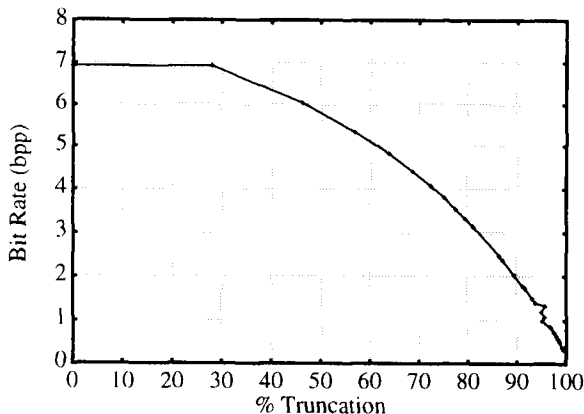


Fig. 6. Effects of the truncation to ZIP compression for lena .img.

The same scheme is also used on another image, camera .img, with almost similar results. Although the quality of the reconstructed image is not as good as that of lena .img, the results show similar trends (see Fig. 7 and 8). Also, Fig. 4 and Table 3 show similar effects of the truncation on the image quality.

Table 3. PSNR values and bit rates of images in Fig. 6.

| % Truncation | PSNR (dB) | Bit Rate (bpp) |
|--------------|-----------|----------------|
| 73 | 40.18 | 3.98 |
| 84 | 31.10 | 2.8 |
| 93 | 23.26 | 1.6 |
| 97 | 19.51 | 0.85 |
| 99 | 17.46 | 0.39 |

3.3. Compression Performance

We can then use the threshold scheme as a simple image compression scheme. The compression takes the forward DWT of the image, truncates the small magnitude coefficients according to a threshold, and losslessly compresses the truncated coefficients to become the

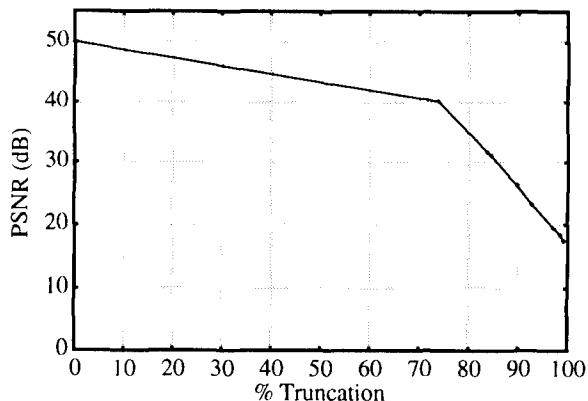


Fig. 7. Effects of % of truncation on the image quality for camera .img.

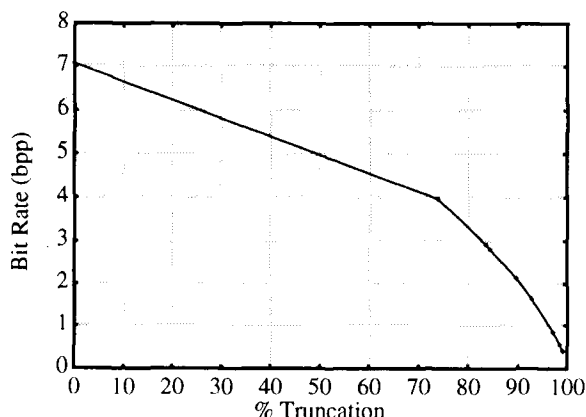


Fig. 8. Effect of the truncation to ZIP compression for camera .img.

compressed image. The decompression then starts with lossless decompression of the compressed image, and inverse transforms the results to obtain viewable image. Figures 9 and 10 show the compression performance measured for lena .img and camera .img, respectively. We observe that in 2 bpp the image still have good quality.

4. DISCUSSION

Localization characteristic and sparsity of wavelet representation are interesting features for image compression. The localized property of the wavelet coefficients reduces the effect of a quantization error to the total image. This is an advantage over Fourier representation. The sparsity means that most of the image energy is distributed among a few basis functions only. Thus a scheme that finely quantizes the high coefficients while coarsely quantizes the small-valued coefficients leads to efficient compression with high quality.

From a practical perspective, the compression scheme is interesting due to the simple and fast computation structure. Since the implementation of the wavelet

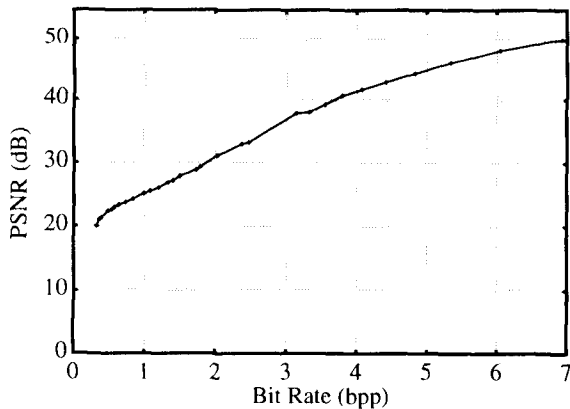


Fig. 9. Quality of the image for different compression rates for `lena.img`.

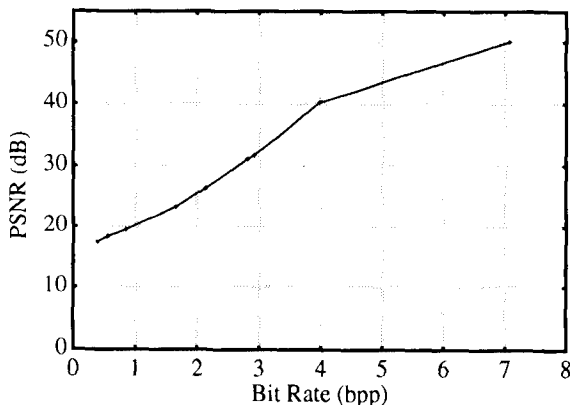


Fig. 10. Quality of the image for different compression rates for `camera.img`.

transform pair consists of filtering only, fast dedicated hardware is possible.

In this scheme, two aspects are open for improvements: (i) selection of the wavelet prototype, and (ii) compression of the coefficients. In our experiment, we selected DAUB4 because of its simplicity, its localized property, and its immediate availability. Thus, the selection has not been made through a study of characteristics of different wavelets. A more elaborate choice of wavelet may lead to much better results. For example, [Mall89] reports the use of image distribution characteristics into consideration results in 1.5 bit/pixel, with only a few noticeable distortions. Furthermore, in our experiment we apply brute force truncation to eliminate redundancy. A more thoughtful method should produce better results. A program which optimally embeds Huffman coding in the compression scheme results in 3:1 lossless compression and 50:1 lossy compression with some degradation [Pres91b].

We conclude that wavelet coding is an important method for image compression. Characteristics of the wavelet representation such as locality and sparsity can

be exploited for compact representation. The magnitude of a wavelet coefficient determines the coefficient's significance with respect to the image fidelity. This can lead to very promising compression schemes as demonstrated in this paper.

ACKNOWLEDGMENTS

This work was supported in part by the Natural Sciences and Engineering Research Council (NSERC) of Canada and the Telecommunications Research Laboratories (TRLabs). One of the authors (AL) wishes to thank IUC-Microelectronics ITB, Laboratory of Signals and Systems ITB, and PT INTI Persero, Bandung, Indonesia, for their support in this research work.

REFERENCES

- [AnRA91] P. H. Ang, T. A. Ruetz, and D. Auld, "Video compression makes big gains", *IEEE Spectrum Magazine*, IEEE Cat. 0018-9235/91, pp. 16-19, Oct. 1991.
- [Daub88] I. Daubechies, "Orthonormal bases of compactly supported wavelets," *Comm. Pure. App. Math.*, v. XLI, 1988, pp. 909-996.
- [Fran69] L. E. Franks, *Signal Theory*. New Jersey: Prentice-Hall Inc., 317 pp., 1969.
- [GoWi87] R. C. Gonzales and P. Wintz, *Digital Image Processing*. Reading MS: Addison-Wesley, 502 pp., 1987
- [JaJS93] N. Jayant, J. Johnston, and R. Safranek, "Signal compression based on models of human perception," *Proceeding IEEE*, 1993, v. 81, no 10, pp. 1385-1421.
- [Kins91] W. Kinsner, "Review of data compression methods, including Shannon-Fano, Huffman, arithmetic, Storer, Lempel-Ziv-Welch, fractal, neural network, and wavelet algorithms", *Technical Report*, DEL91-1, University of Manitoba, 157 pp., Jan. 1991.
- [Mall89] S. Mallat, "A theory for multiresolution signal decomposition: the wavelet representation," *IEEE Trans. Patt. Anal. Machine Intell.*, vol. 11, no 7, pp. 84-95, July 1989.
- [Prag92] D. S. Prague, "How will multimedia change system storage ?", *BYTE Magazine*, McGraw-Hill, Peterborough NH, pp. 164-165, Mar. 1992.
- [Pres91a] W. H. Press, Wavelet Transforms, *Harvard-Smithsonian Centre for Astrophysics Preprint No. 3184*, 1991. (Available through anonymous ftp at 128.103.40.79, directory /pub).
- [Pres91b] W. H. Press, FITSPRESS, *Harvard-Smithsonian Centre for Astrophysics*, Beta Version 0.8, 1991. (Available through anonymous ftp at 128.103.40.79, directory /pub and file fitspress08.tar.z).
- [RiVe91] O. Rioul and M. Vetterly, "Wavelets and signal processing", *IEEE Signal Processing Magz.*, IEEE CT 1053-5888/91 pp. 14-38, Oct. 1991.

Negative Frequencies and Complex Signals

*Yes, there are such things as negative frequencies.
Understanding them helps us analyze and process signals.*

by Jon Bloom, KE3Z

The concept of a negative frequency is one that seems strange to many amateurs. Our radios don't have negative frequency values engraved on their dials, after all. But negative frequencies do exist. In this article, I hope to convince you of that and to extend the discussion to show how representing signals as complex numbers eases signal processing.

We run into the problem of negative frequencies with mixers. We are told in our basic radio texts that feeding signals into the two input ports of a mixer produces output signals at the sum and difference of the two frequencies. But if we calculate those output frequencies, we find that we can get negative values of frequency as well as positive values. Usually, the text either ignores these negative values or dismisses them as unreal. But they aren't unreal. In fact, we will find that for every positive frequency coming out of the mixer there is a corresponding negative frequency present. Why?

We need to begin at the beginning. The basic signal we'll work with is a sine wave. Not only is this signal a convenient one, it turns out that any real signal can be analyzed as a number of sine waves added together. It's important to keep the "can be analyzed" part in mind. What we are doing here is attempting to represent actual signals as mathematical expressions. If that seems abstract, removed from the real world, consider that our use of the word "frequency" is just as abstract. Frequency is not a physical quantity like mass or charge. It is a rate—the rate at which a signal repeats itself. In other words, it is a particular mathematical characteristic of a signal.

Representing Sine Waves

A sine wave is formed by rotating a constant-amplitude vector around a point. This is shown in Fig 1. It's most convenient to place the point at the origin of a Cartesian plane—a set of x - y axes. The amplitude of the sine wave can be found by determining, for a given point in time, the y displacement of the end of the vector. If a cosine wave is wanted, the x displacement is used. Throughout this article, we'll use co-

sine waves for convenience. ("Sine wave" is really a generic term; it includes sinusoidal waves of any phase: sine, cosine or whatever.)

What we've done by using a rotating vector is to create a model for generating or analyzing a sine wave. We can use this model to find a mathematical expression—a function—for the sine wave. Once we have that, we can process the signal with math operations to change it to be what we want it to be. If our processing is done in analog

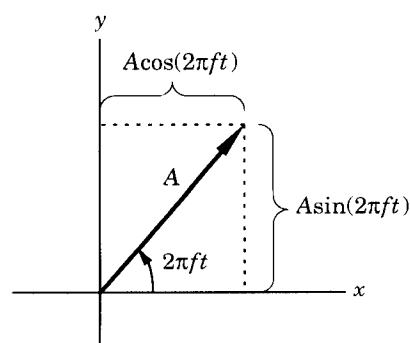


Fig 1—A sine wave is formed by a rotating vector that has x and y components.

electronics, the math operations are implemented as circuits: a linear amplifier to multiply the signal function by a constant value; a voltage divider to divide by a constant value; a mixer to multiply two signals together; a circuit containing reactances to cause delayed (phase-shifted) components of the signal to add to and subtract from one another. If we're processing the signal in a DSP system, we of course can perform the needed math in a computer. But however we physically do the processing, we are manipulating the signal using mathematics. The point is this: once we choose to apply math to do our signal processing, we have to take care that the math is complete and consistent. That means that if the math shows negative frequencies to be part of the signal, we can't afford to ignore them. And, as it turns out, recognizing the existence of these negative frequencies leads to a more intuitive understanding of what happens in some kinds of signal processing, like mixing.

From the rotating vector of Fig 1, we can write a math function that expresses the signal amplitude at any point in time. The vector rotates at a constant rate, making some number of circles (cycles) per second—the frequency, f . In each of those cycles, the vector rotates through 360° , or 2π radians. The angular rate of rotation of the vector is therefore $2\pi f$ /radian/s. To find the angle of the vector at any point in time, all we need to do is to multiply the rate by the time: $2\pi ft$. The signal amplitude of a cosine wave is the x component of the vector, which from trigonometry is the length of the vector times the cosine of the angle: $A\cos(2\pi ft)$. To complete the expression, we write it as an equation:

$$a(t) = A \cos(2\pi ft) \quad \text{Eq 1}$$

In this case, a is the instantaneous amplitude of the signal. The expression $a(t)$ is read as “ a as a function of t ,” and simply shows that plugging a particular value of t into the equation results in a particular value for a .

In Eq 1, the value of $2\pi f$ is constant (we assume the frequency is constant). So, rather than write $2\pi f$ in all of our equations, we often use the identity:

$$\omega = 2\pi f \quad \text{Eq 2}$$

Which changes Eq 1 to:

$$a(t) = A \cos(\omega t) \quad \text{Eq 3}$$

This is a fine way of representing the sine-wave signal mathematically—we can take any value of t and find both the angle of the vector and the corresponding value for a . But there's a

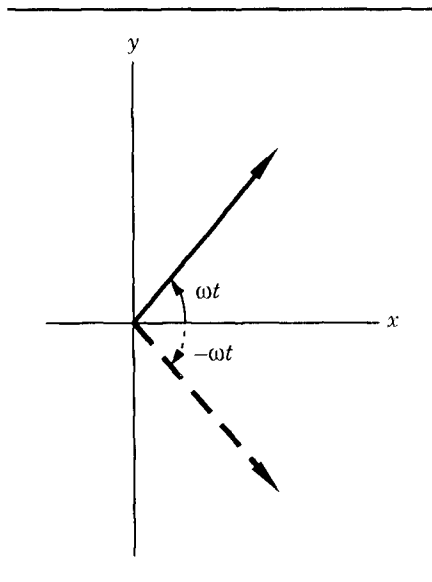


Fig 2—Knowing only the cosine of the vector's angle is insufficient; there are two angles that have the same cosine value.

problem: we can't take a value of a and find the corresponding angle of the vector. Fig 2 shows why: there are *two* possible angles that produce the same value for a . If the amplitude could have been produced at an angle of ωt , it could as easily have been produced at an angle of $-\omega t$. That means that we can't determine the phase of the sine wave from its amplitude. That will be a problem if we do signal processing in which phase is important; it will be difficult to find math operations that do what we want, all because our original rotating-vector model is incomplete. It's adequate for generating the signal, because we can get a precise, unambiguous value of a for any angle. But it's not suitable for completely analyzing the signal because we can't derive an unambiguous angle based on a particular value of a .

What we need to complete the model is more information—a second variable, to remove the ambiguity between the two possible angles that can produce the instantaneous amplitude. But we don't *have* more information; all we have is the amplitude. We'll see later that there is a way of developing more information, but for now, we don't have it. So, we need a way of removing the ambiguity from our model. We can do that by adding a second vector. If the sine-wave signal is composed of two vectors, one at a positive angle and one at a negative angle, as shown in Fig 3, our problem is solved. Now, for any value of a we can determine the angles of the two vectors. In

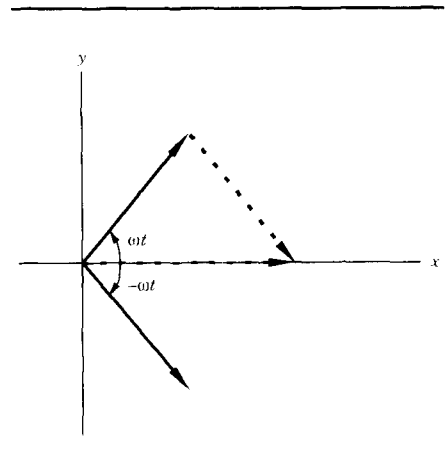


Fig 3—A real-number signal can be considered to be composed of two rotating vectors, shown as solid lines. The sum of those two vectors, shown with dotted lines, always lies on the x axis.

this case, while one vector is rotating clockwise, the other is rotating counterclockwise, so that its angle will always be the negative of the first vector's. The rotational speed of the clockwise-rotating vector is ω . The other vector must then have a speed of $-\omega$. In other words, its frequency is negative. Since each vector contributes its x component to the amplitude of the signal, the length of each vector has to be half of the length of the original single vector of Fig 1. The equation that expresses this new two-vector model is:

$$a(t) = \frac{A}{2} \cos(\omega t) + \frac{A}{2} \cos(-\omega t) \quad \text{Eq 4}$$

This new equation solves our problem; from any value of a we can find the angles of the two rotating vectors. What Eq 4 shows is that the signal is composed of two frequency components, a positive one and a negative one. This is characteristic of any signal whose amplitude is composed of real-number values. Note that the sum of the two vectors will always lie on the x axis: it's a real number.

If you are still skeptical about the reality of negative frequencies, consider this: in physics, we often talk about negative velocity. What is meant by this is not that the speed of the item is less than zero, but that the direction is opposite to whatever direction we called positive. The same is true here. A negative frequency doesn't mean that the signal repeats itself less than 0 times per second, but that the vector is rotating in the opposite direction from that of a positive-frequency signal.

Most useful signals are not a sine wave, but something more complicated. Some mathematics can be used—although we won't go through it here—to show that such signals can be considered to be the summation of a number of sine waves of varying frequencies, phases and amplitudes.

Each of the sine waves in the signal has both positive and negative frequency components, just like the sine wave we have discussed.

Mixing Signals

Mixing is an integral part of much of the signal processing we do. The ideal

double-balanced mixer would multiply each instantaneous amplitude value of one of its input signals by the corresponding (in time) amplitude of the other input signal to produce the output signal. If the two input signals are sine waves of different frequencies, one at ω_1 and one at ω_2 , the result of the mixing is:

$$y(t) = A \cos(\omega_1 t) \cdot B \cos(\omega_2 t)$$

Here, y is the output signal from the mixer.

That's fine, but what does that mean in terms of the frequency content of y ? The mathematical answer is that multiplying two time-varying signals can be analyzed in the frequency domain by means of *convolution*. We could take several pages here to explain convolution, but there is a simpler way of looking at the process: Each frequency component of one input signal shifts all of the frequency components of the other signal. Fig 4 shows this for our example mixing of two sine waves. We'll choose $\cos(\omega_1 t)$ as the signal to be shifted, and $\cos(\omega_2 t)$ as the signal that does the shifting. The spectrum of the first signal is shown in Fig 4A; it consists of two components, at frequencies of ω_1 and $-\omega_1$. Similarly, the second signal, Fig 4B, has components of ω_2 and $-\omega_2$. Fig 4C shows the output after mixing. The $\pm\omega_1$ components have been shifted up in frequency by the ω_2 component, resulting in components at $(\omega_2 + \omega_1)$ and $(\omega_2 - \omega_1)$. And the $\pm\omega_1$ components have also been shifted down in frequency by the $-\omega_2$ component, resulting in $(-\omega_2 + \omega_1)$ and

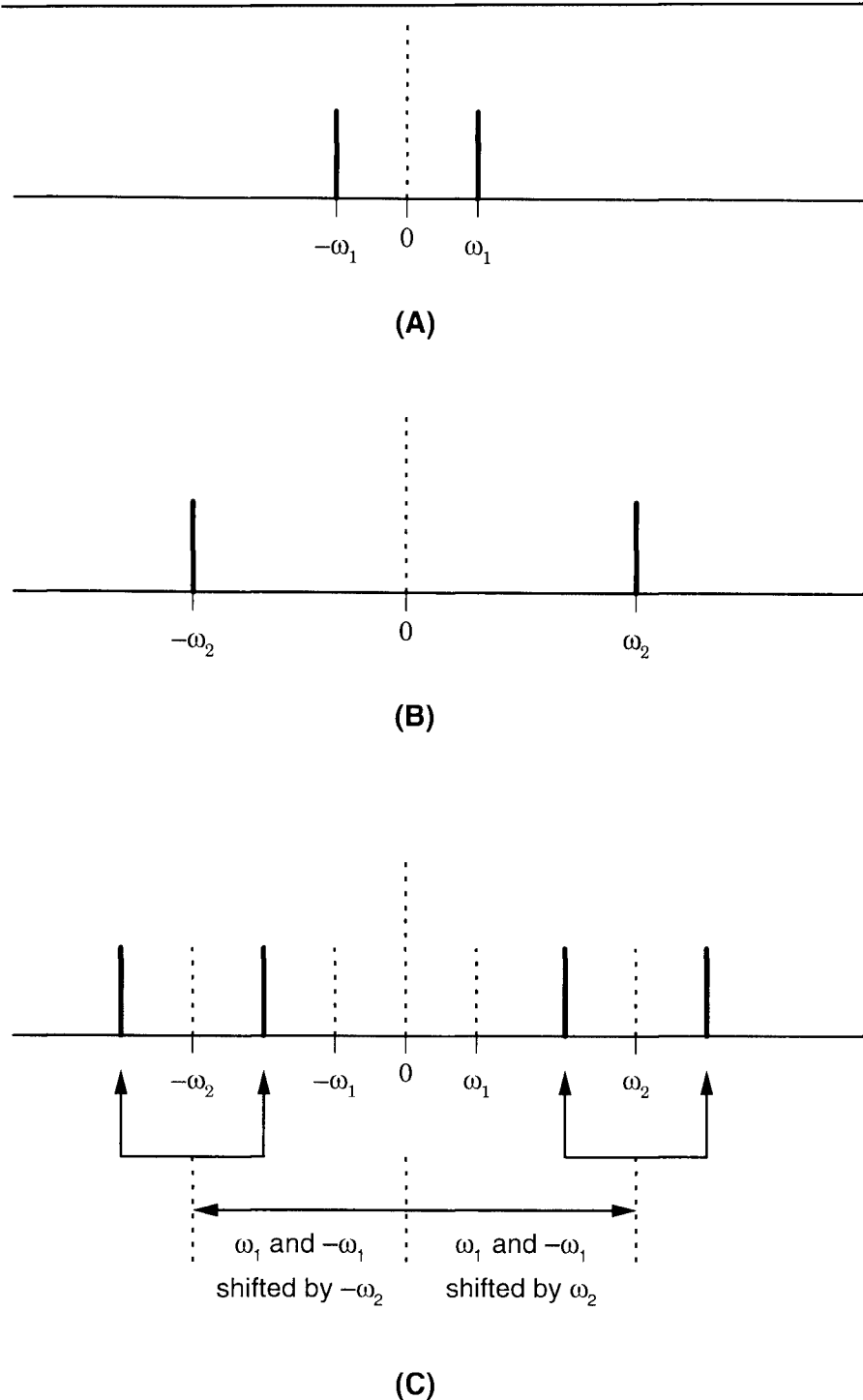


Fig 4—Mixing two real-number signals. The spectral components are shown on a frequency-versus-amplitude graph. The two inputs to the mixer, $\pm\omega_1$ and $\pm\omega_2$, are at A and B, respectively. The result of the mixing is shown at C.

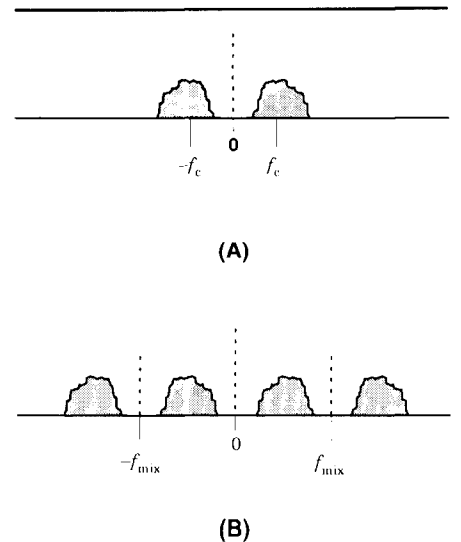


Fig 5—Mixing a signal composed of a carrier (at f_c) and its sidebands with a single-frequency signal (at f_{mix}).

$(-\omega_2 - \omega_1)$ components. We could have chosen the ω_2 signal as the one to be shifted and the ω_1 signal as the one to do the shifting; we would have gotten the same result. (Try it!)

Fig 5 shows the mixing process when one of the input signals consists of a number of frequency components. For example, a modulated signal, which consists of the carrier and its surrounding sidebands. In this drawing, we're representing this signal by showing a band of frequencies that the signal might occupy. Mixing this signal with a sine-wave signal results in the output shown in Fig 5B.

This way of looking at mixing seems more intuitive than the way most amateurs originally learned. ("Mixing produces sum-and-difference products...") And, as we'll see, it is easily extended to complex signals.

Complex Signals

The preceding discussion shows what happens when we're dealing with real sine waves that have a single amplitude value at each point in time. In our model, we had to resort to two rotating vectors—two frequencies—to make it work. If only we had an additional piece of information at each point along the waveform of our sine wave, we could find the angle of our single rotating vector (Fig 1) from the instantaneous amplitude unambiguously. There are several candidates for this additional variable. For example, we could use the vector's angle itself. If our signal carried not only amplitude information but angle information as well, we'd be all set. In our processing we would need to have two values for each point in the waveform—two channels in the analog electronics, or two sets of numbers in a digital computer.

Angle values would work, but a more convenient second variable can be found in Fig 1. If the signal amplitude values were to tell us not only the cosine of the vector's angle, but also the sine of that angle, there is only one possible angle that matches the signal's instantaneous value. Including the sine value removes the angle ambiguity because the two angles of Fig 2 that share the same cosine value have different sine values. So a sine-wave signal that consists of two values at each point of the waveform—the cosine value and the sine value—describes a single rotating vector.

Since we want to process our signals mathematically, it would be nice to find a way of combining the two values

that make up a point of the waveform into a single number. We do that by using the two values to form a complex number, which we can do because the cosine and sine components are at right angles to one another, just as are the real and imaginary parts of a complex number. Now, our signal is expressed so:

$$a(t) = A \cos(\omega t) + jA \sin(\omega t) \quad \text{Eq 5}$$

with the j being $\sqrt{-1}$.

Since Eq 5 describes a single vector, there is only one frequency component. As written, the equation describes a vector rotating in the clockwise direction; it has a positive frequency. A negative-frequency vector would be rotating in the counter-clockwise direction, resulting in the equation:

$$a(t) = A \cos(\omega t) - jA \sin(\omega t) \quad \text{Eq 6}$$

Just as in the case of signals made up of real-number amplitude values, complex-number signals can consist of multiple sine waves. In this case, the signal need not have both positive and negative frequency components, although it may. In fact, it may have a positive component (or components) at frequencies different from those of the negative component (or components).

If a complex signal contains only positive-frequency components, or only negative-frequency components, it's called an *analytic* signal.

In our signal processing system, we call the signal path of the real part of the complex numbers the *in-phase* (I) channel. The signal path of the imaginary part is called the *quadrature* (Q) channel.

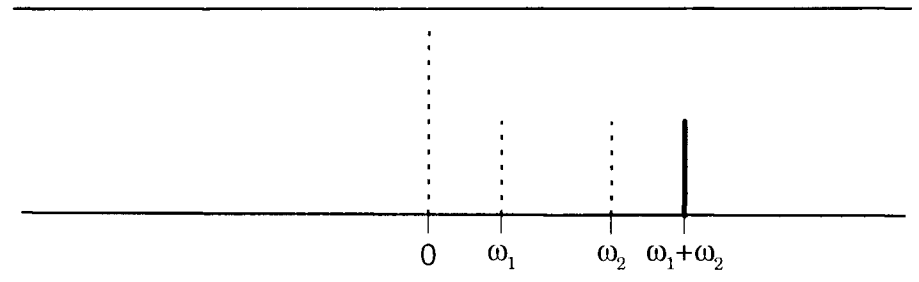


Fig 6—Mixing two analytic sinusoids results in a simple frequency shift.

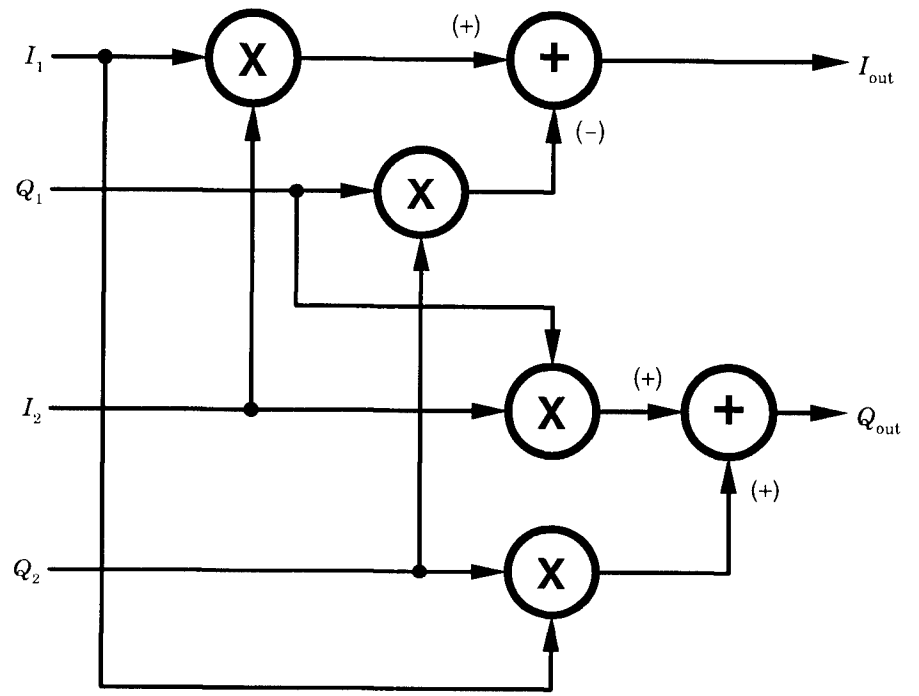


Fig 7—Multiplying two complex signals (I_1, Q_1 and I_2, Q_2) requires four real multiplications and two real additions.

Mixing Complex Signals

When we mixed two real-number signals, we said that each frequency component of the shifting signal moves each component of the shifted signal by an amount equal to the shifting component. That's still true if the two signals we're mixing are complex. For example, if we mix two complex signals, each of which contains a single positive-frequency component, the result is a signal whose frequency is the sum of the two input signal frequencies; one of the signals has shifted the other up, as shown in Fig 6. Or, if the shifting signal was a single negative frequency, the other signal would be shifted down.

This shows the utility of viewing mixing as a frequency-shifting process. It doesn't matter what the frequency components of the signals are, or whether they are real or complex; the shifting mechanism is the same in all cases.

Of course, multiplying two complex signals is a bit more difficult than multiplying two real signals. Each complex signal value consists of a real part and an imaginary part. Multiplying the two numbers becomes:

$$(a + jb)(c + jd) = (ac - bd) + j(bc + ad) \quad \text{Eq 7}$$

Since the real and imaginary parts are handled in separate channels in our signal-processing hardware or software, implementing Eq 7 requires something like the block diagram of Fig 7. This isn't something you want to do in analog electronics if you can avoid it! In DSP, however, it's not difficult, although repeated complex multiplications may require more processor power than we'd like.

Creating Analytic Signals

Often, we want to process a real-number signal, but we'd like the flexibility of an analytic signal, without the "mirror image" negative-frequency components. So, we would like to create a signal that has only the positive-frequency components of the input. To convert the input signal into an analytic signal, we will need to create the second set of values, which will become the imaginary part of our complex numbers. The question is, how?

Consider once again the rotating sine-wave vector of Fig 1. We have as our input signal the x , or cosine, component of this vector. We need the sine component as well for our analytic signal. But this is a sine wave. The sine

component at a particular time is equal to the cosine component of a quarter cycle (90°) earlier. To put it another way, the sine component is equal to the cosine component, phase shifted by 90° . So, if we could generate a signal that is like the input signal, but with every frequency component shifted by 90° , we could use that as the imaginary (quadrature) part of our complex signal.

Generating this quadrature signal can be difficult, especially in analog electronics. Every frequency component of the signal must be shifted by 90° . That means that each of these components must be delayed by a quarter cycle. But since the period of each frequency component is different, the needed delay is different. This isn't a big problem for a narrow-band, high-frequency signal. For example, for a 3000-Hz-wide signal centered at 9 MHz, all of the components within the signal are at nearly the same frequency, so a fixed delay of:

$$\frac{1}{4} \left(\frac{1}{9 \text{ MHz}} \right) = 27.8 \text{ ns} \quad \text{Eq 8}$$

provides very nearly a 90° shift for all of the signal frequencies. (The phase error is about 0.02° .) But have that same 3-kHz bandwidth centered at 10 kHz, and a fixed delay won't work. In that case, you have to resort to some pretty complex analog circuitry to achieve the needed phase shift.

In DSP, we have a tool, called the *Hilbert transformer*, that provides a 90° phase shift of a broad-band signal. Hilbert transformers are implemented as finite-impulse-response (FIR) filters, which are reasonably efficient DSP algorithms. That's one reason why use of complex signals is more common in DSP than in analog electronics.

Half-Complex Mixers

There is another way of turning a real-number signal into a complex signal, a way that is more practical in analog implementations. The scheme is shown in Fig 8. Here, a real input signal is mixed separately with both the real and imaginary components of an single-frequency analytic signal. This causes both the positive- and negative-frequency components of the real signal to be shifted up (in this case) in frequency. If the negative-frequency component is shifted up enough to become a positive-frequency component, the resulting signal contains only positive-frequency components: it's an analytic signal. This scheme is known as a *half-complex mixer*.

Complex Signals and the Weaver Method

The Weaver method of generating and demodulating SSB signals is well known.^{1,2} It's useful, though, to con-

¹Notes appear on page 27.

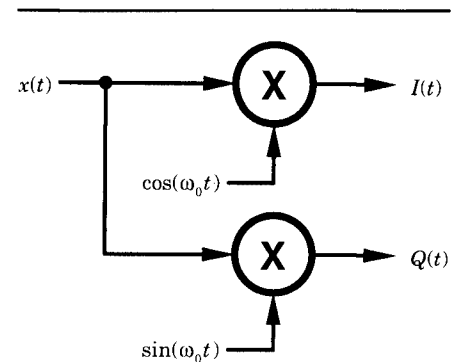


Fig 8—A half-complex mixer creates a frequency-shifted complex signal from a real-number signal.

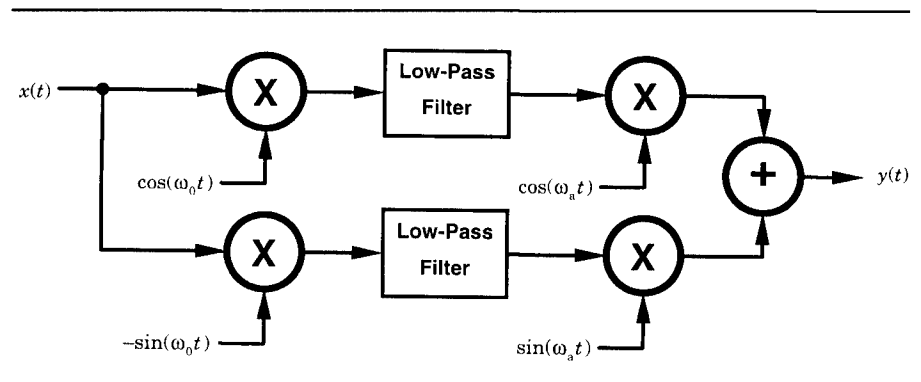


Fig 9—Block diagram of the Weaver method of demodulating SSB.

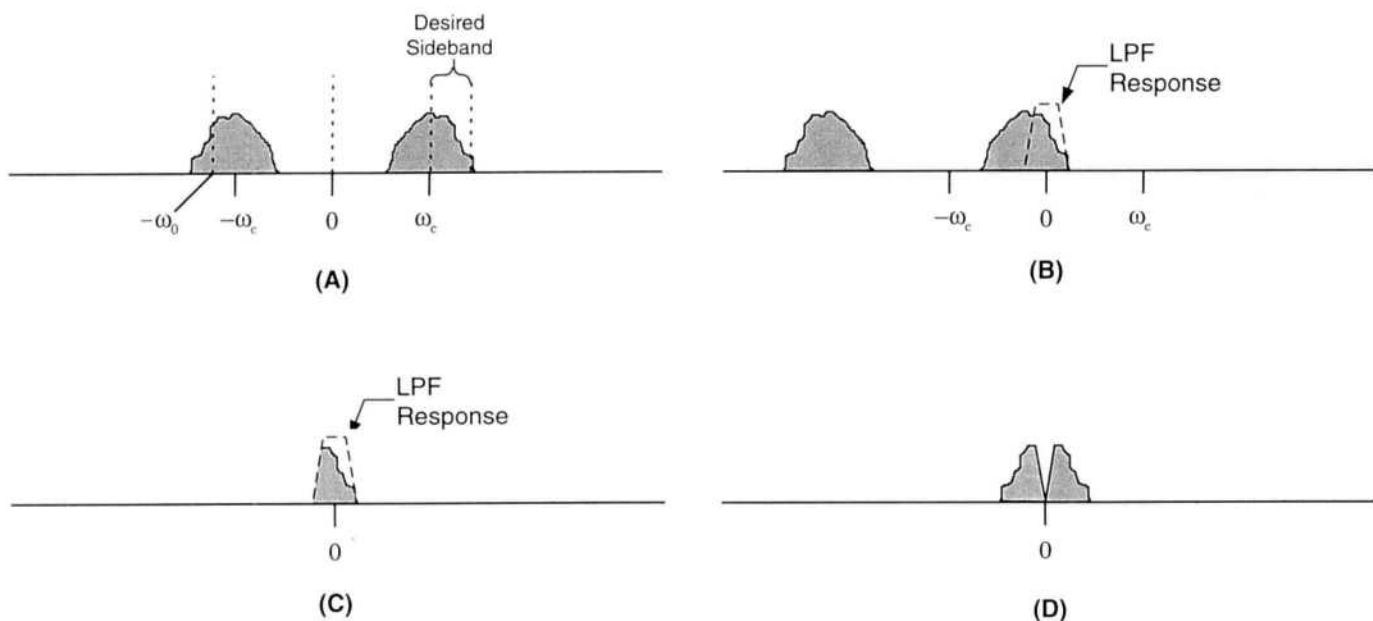


Fig 10—Steps in the processing of an SSB signal via the Weaver method of Fig 9. The input signal is at A. It is first shifted down in frequency via a half-complex mixer, resulting in the spectrum at B. Low-pass filters remove the unwanted spectral components—everything except the desired sideband—to produce the signal at C. This complex signal is then shifted up (multiplied by a single-frequency analytic signal) so that the signal is in the proper audio frequency range, with only the real part of the result (D) used as the output of the system.

consider this technique as one using complex signals. The basic scheme is shown in Fig 9. Here, the incoming signal is mixed with an analytic signal at a negative frequency ($-\omega_0$) in the center of the desired sideband. This shifts the input signal down in frequency so the center of the sideband is at 0 Hz, as shown in Fig 10B. Low-pass filters then eliminate the unwanted sideband and any adjacent-channel signals. The resulting complex signal is then mixed with an analytic signal at a positive frequency (ω_a) equal to the center of the audio range of the modulation, which shifts the signal up to the proper audio frequencies.

You can analyze this method using trigonometry, as has been done in numerous publications. But how much easier it becomes when you recognize the use of complex signals in the mixing process! The first set of mixers in Fig 9 form a half-complex mixer like that of Fig 8. The second set of mixers operate like the complex mixer of Fig 7, except that only the real-number part of the result is computed; we only need that part to output the audio signal.

Conclusion

It's reasonable to wonder why, since negative frequencies exist, we don't often hear about them in analog elec-

tronics. The reason is that, since any real signal contains negative-frequency components equal to its positive-frequency components, we can afford to neglect them. They are there, but we can simplify much of our view of analog electronics by using such maxims as: "a mixer produces signals at the sums and differences of the input frequencies." That simplified view is useful, but it isn't the whole story. And once we begin doing more complicated processing of the signal, use of complex signals makes that simplification unusable, since we no longer are assured that the negative-frequency spectrum will mirror the positive frequencies.

Negative frequencies and complex signals are a valuable analysis tool for use in understanding the operation of signal processing systems. Particularly in the case of DSP implementations, but also in some analog electronics, they provide a simple, intuitive way of looking at some otherwise nonintuitive processes.

Notes

¹Puig, Carlos M., KJ6ST, "A Weaver Method SSB Modulator Using DSP," *QEX*, September, 1993, pp 8-13.

²Anderson, Peter Traneus, KC1HR, "A Different Weave of SSB Receiver," *QEX*, September, 1993, pp 3-7. □

CAPMANtm

Computer Assisted Prediction Manager

SKYWAVE ANALYSIS

"Use IONCAP⁺ like a pro!"

Package includes:

CAPMAN - Menu driven input & control with context-sensitive help!
Multicolor - Output Graphics display!
Extra or Novice, its EASY to use!

IONCAP⁺ - the same 32 bit tool the pros use to get S/N, MUF, FOT, LUF, Reliability and much More! Special Long-distance model for DXers!

MININEC & ELNEC interface for your custom antenna input!

- + Customize all input parameters for your station!
- + Your contacts and friends may be added to the library and run any time with a few keystrokes.
- + A full featured location database, indexed on country name and call prefix, provides access to over 490 prefixes for quick one-step predictions.
- + The package includes the newest "updated" full commercial version of IONCAP used by over 450 government agencies and commercial depts. in the USA and over 100 other countries.
- + Many more features in a powerful "HF Analysis" package at a fraction of the commercial price.

Entire CAPMAN package — \$89
USA postage paid. Overseas add \$3.50

Order today from:

LUCAS Radio/Kangaroo Tabor Software
2900 Valmont Rd, Suite H
Boulder, CO 80301

(303) 494-4647 (494-0937 fax)

VISA MC CHECK MONEY ORDER

RF

by Zack Lau, KH6CP/1

A Low-Noise PHEMT Amplifier for 5760 MHz

With the growing interest in the 6-cm band, for both terrestrial and satellite work, I've decided that this band could use more circuits to copy. Unlike at VHF, where the sky is pretty noisy, cold-sky temperatures can be

quite low on this band, sometimes getting close to the 3-Kelvin limit associated with the Big Bang. Because of the low sky temperature, a 0.7-dB system noise figure can result in as much as a 7-dB improvement over a 3-dB system. In contrast, a 0.7-dB system noise figure at 2 meters is likely to have only a 2-dB improvement over a 3-dB system. As amateurs in heavily populated areas have found, even this much of an improvement may not be available at

2 meters because of the high noise level from electrical interference on VHF. The design presented here has a 0.65-dB noise figure with 22 dB of gain. This noise figure is approximately 0.3-dB less than that realized from good designs that use the ATF 10135 GaAs FET.

225 Main Street
Newington, CT 06111
Email: zlau@arrl.org (Internet)

System Design Considerations

While the goal is to minimize the noise figure of the preamp as much as

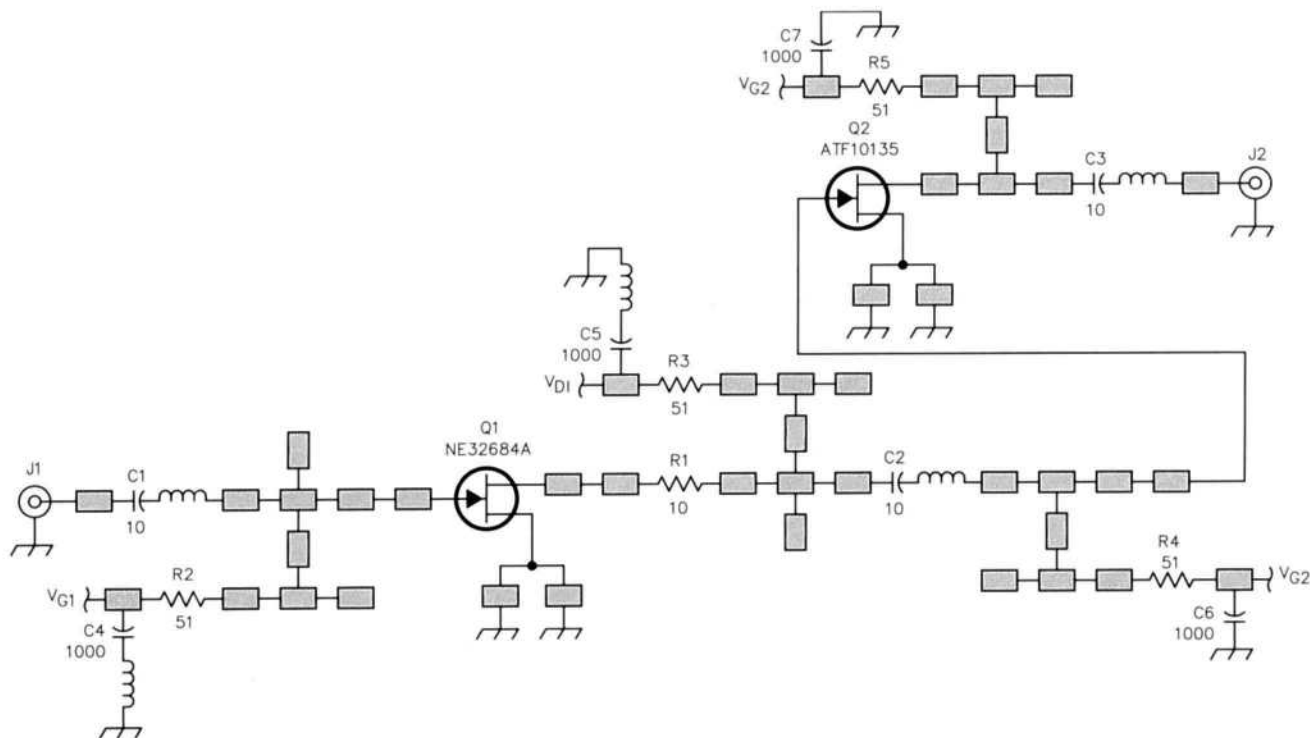


Fig 1—Schematic diagram of the 5760-MHz, two-stage, low-noise preamplifier.

C1-C3—10-pF ATC 100A chip capacitors.
C4-C7—1000-pF chip capacitors.
J1, J2—SMA panel jacks.
Q1—NEC 32684A PHEMT, available from California Eastern Laboratories. Long-lead device preferred.

Q2—ATF 10135 GaAs FET, available from Hewlett Packard. Short-lead devices are usable (see text for details).
R1—10- Ω chip resistor.
R2-R5—51- Ω chip resistors.

Microstrip transmission lines are indicated in the drawing by rectangles. These are modeled in the *Microwave Harmonica* simulation of the circuit, as are the stray (unlabeled) inductances shown.

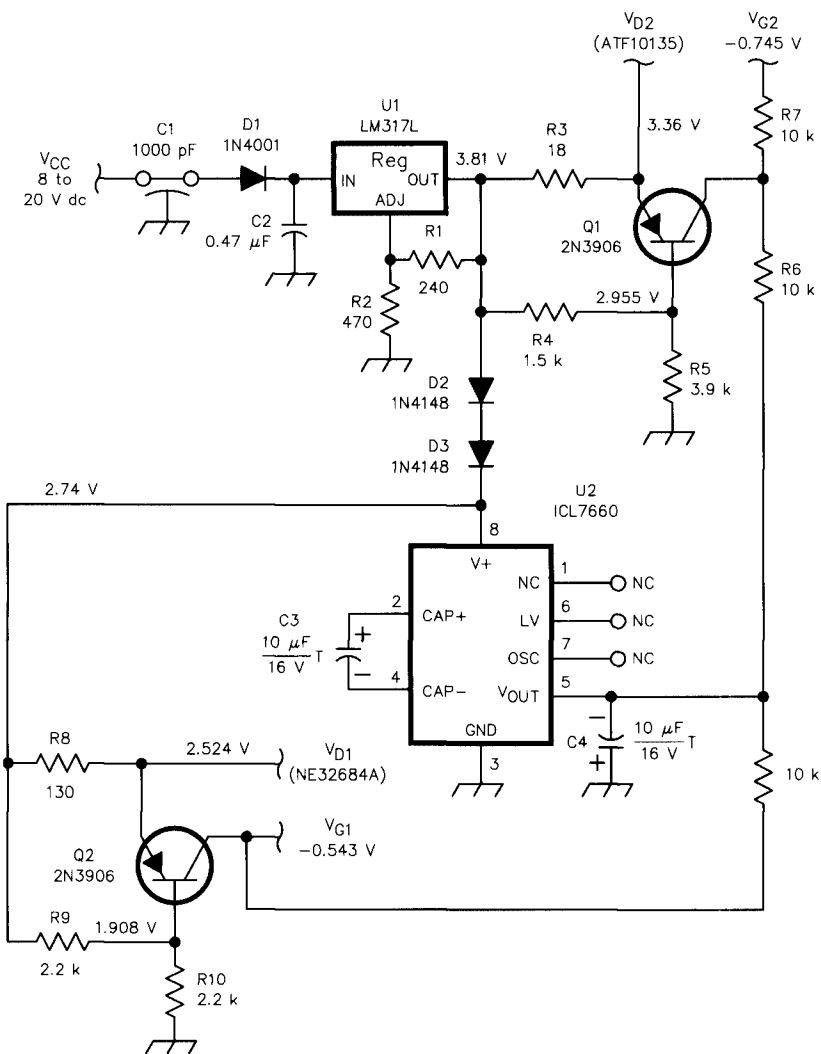


Fig 2—Bias supply circuit for the GaAs FETs.

C1—Feedthrough capacitor. The value is not critical.

Q1, Q2—2N3906, 2N2907, or other general-purpose PNP transistors.

U1—LM317L adjustable voltage regulator IC.

U2—Intersil ICL7660CPA or SI7660CPJ CMOS voltage converter IC.

possible, it is often wise to figure out how the preamp fits into the system. Then, when you fit everything together, you should get the theoretical performance you calculated with a minimum of tweaking.

Perhaps the most important factor is stability. If your preamp oscillates, you are better off without it in your system. Some people actually get it backward; they only consider stability at the design frequency. Actually, the design frequency is where you have the most flexibility. After all, amateurs typically take great pains to match their antenna to the transmitter, so the impedance of the antenna is often pretty close to 50 Ω, at least on the band you are using. So, conditional stability, or stability with good terminations, may be all you need at the design frequency. (This assumes that the second stage has a well behaved input impedance, like that of a broadband MMIC.) On the other hand, impedances can get pretty wild once you get out of band, so your preamp should be unconditionally stable outside the design frequency range.

The gain of the amplifier is another important consideration; the noise figure of a microwave LNA doesn't necessarily set the *system* noise figure. If you do a few calculations, considering

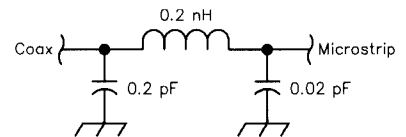


Fig 3—Model of the SMA-to-microstrip transition.

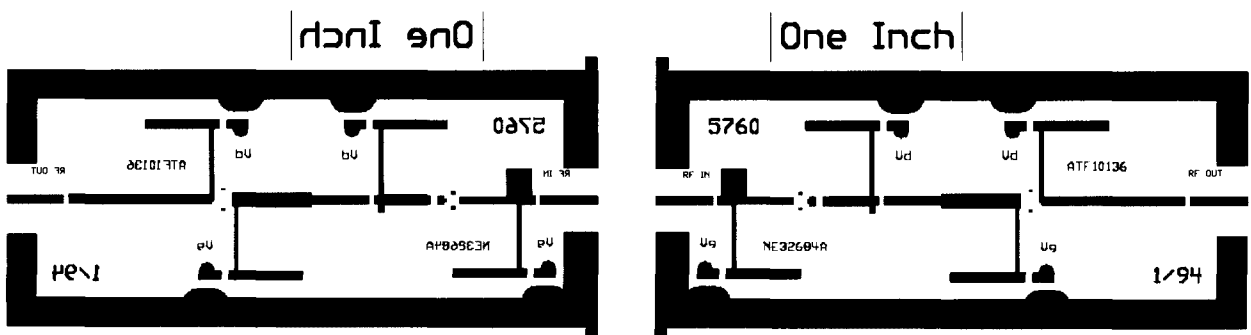


Fig 4—Etching pattern for the preamplifier circuit board. Use 15-mil 5880 Duroid.

Table 1—Microwave Harmonica Output Listing

| Freq (GHz) | NF (dB) | | MS11 (dB) | | MS21 (dB) | | MS22 (dB) | | K | | NF (dB) | | MS11 (dB) | | MS21 (dB) | | MS22 (dB) | | K | |
|---------------|---------|--------|-----------|--------|-----------|-------|-----------|--------|--------|--------|---------|--------|-----------|--------|-----------|------|-----------|------|------|------|
| | LNZ | LNZ | LNZ | LNZ | LNZ | LNZ | LNZ | LNZ | LNZ | LNZ | LNZ | AMP1 | AMP2 | AMP1 | AMP2 | AMP1 | AMP2 | AMP1 | AMP2 | AMP1 |
| 0.50 | 7.05 | -15.79 | 3.79 | -29.64 | 184.66 | 7.78 | -8.61 | 3.73 | -10.40 | 44.05 | 8.35 | -15.80 | 8.12 | -10.42 | 999.90 | | | | | |
| 1.00 | 6.24 | -10.50 | 5.04 | -21.42 | 63.23 | 6.60 | -9.00 | 4.92 | -20.25 | 18.86 | 7.21 | -10.54 | 10.28 | -20.05 | 999.90 | | | | | |
| 1.50 | 5.63 | -7.03 | 5.83 | -16.12 | 30.77 | 5.77 | -7.44 | 5.17 | -25.74 | 10.69 | 6.36 | -7.04 | 11.27 | -25.18 | 772.91 | | | | | |
| 2.00 | 5.46 | -5.24 | 5.64 | -12.89 | 20.31 | 5.58 | -6.95 | 4.47 | -29.42 | 8.46 | 6.14 | -5.13 | 10.89 | -28.92 | 376.64 | | | | | |
| 2.50 | 5.67 | -4.53 | 4.76 | -11.13 | 17.35 | 6.13 | -7.98 | 2.43 | -14.24 | 10.11 | 6.66 | -4.44 | 7.96 | -13.78 | 375.61 | | | | | |
| 3.00 | 5.68 | -4.76 | 5.17 | -10.59 | 12.97 | 6.85 | -12.57 | -0.09 | -9.17 | 14.66 | 7.19 | -4.72 | 4.42 | -8.97 | 506.78 | | | | | |
| 3.50 | 3.05 | -2.75 | 11.40 | -11.81 | 1.93 | 3.31 | -14.96 | 6.05 | -10.42 | 3.24 | 3.25 | -2.53 | 17.07 | -9.51 | 12.94 | | | | | |
| 4.00 | 1.73 | -3.10 | 13.39 | -17.28 | 1.26 | 1.23 | -7.19 | 10.16 | -13.55 | 1.19 | 1.78 | -4.24 | 23.24 | -11.17 | 3.28 | | | | | |
| 4.50 | 1.32 | -4.31 | 13.95 | -42.51 | 1.21 | 1.03 | -6.32 | 10.10 | -10.51 | 1.06 | 1.35 | -4.59 | 24.07 | -10.13 | 1.98 | | | | | |
| 5.00 | 0.98 | -6.48 | 14.35 | -18.10 | 1.21 | 0.95 | -7.03 | 9.91 | -10.66 | 1.06 | 1.01 | -4.08 | 24.53 | -10.13 | 1.53 | | | | | |
| 5.50 | 0.68 | -13.37 | 14.74 | -16.73 | 1.23 | 0.86 | -12.42 | 10.13 | -20.79 | 1.06 | 0.71 | -16.24 | 25.04 | -26.48 | 1.71 | | | | | |
| 5.76 | 0.65 | -20.96 | 14.61 | -19.40 | 1.25 | 0.88 | -19.03 | 10.01 | -20.12 | 1.06 | 0.69 | -23.25 | 24.32 | -19.60 | 1.79 | | | | | |
| 6.00 | 0.80 | -10.27 | 13.87 | -15.08 | 1.28 | 0.97 | -14.74 | 9.56 | -12.13 | 1.06 | 0.83 | -13.57 | 23.69 | -17.65 | 1.78 | | | | | |
| 6.50 | 1.84 | -2.86 | 10.00 | -6.08 | 1.37 | 1.35 | -8.32 | 8.15 | -8.60 | 1.08 | 1.98 | -2.45 | 17.97 | -5.31 | 2.04 | | | | | |
| 7.00 | 3.75 | -1.18 | 5.10 | -3.29 | 1.50 | 2.02 | -7.07 | 7.16 | -12.92 | 1.10 | 4.37 | -1.03 | 11.45 | -12.18 | 4.06 | | | | | |
| 7.50 | 5.89 | -0.71 | 0.84 | -2.19 | 1.64 | 2.90 | -4.64 | 5.94 | -14.42 | 1.11 | 7.18 | -0.67 | 5.89 | -8.45 | 7.62 | | | | | |
| 8.00 | 7.84 | -0.55 | -2.58 | -1.70 | 1.83 | 3.44 | -2.51 | 3.88 | -5.62 | 1.13 | 9.07 | -0.59 | 2.07 | -3.54 | 8.70 | | | | | |
| 8.50 | 9.45 | -0.50 | -5.30 | -1.43 | 2.10 | 2.83 | -2.78 | 2.99 | -4.13 | 1.22 | 9.98 | -0.61 | 5.04 | -6.58 | 4.83 | | | | | |
| 9.00 | 10.84 | -0.50 | -7.71 | -1.26 | 2.66 | 3.43 | -6.25 | 1.89 | -4.75 | 1.75 | 14.18 | -0.47 | -7.37 | -3.80 | 45.31 | | | | | |
| 9.50 | 13.04 | -0.65 | -11.82 | -1.04 | 6.60 | 14.02 | -1.97 | -13.85 | -3.88 | 21.03 | 27.87 | -0.66 | -27.77 | -3.84 | 999.90 | | | | | |
| 10.00 | 9.29 | -1.50 | -7.04 | -1.75 | 7.68 | 21.31 | -1.33 | -24.91 | -1.89 | 109.64 | 23.17 | -1.51 | -26.53 | -1.90 | 999.90 | | | | | |
| 10.50 | 9.72 | -0.68 | -4.50 | -2.60 | 2.32 | 7.75 | -2.18 | -8.29 | -2.35 | 4.16 | 11.57 | -0.51 | -7.98 | -2.12 | 25.43 | | | | | |
| 11.00 | 9.62 | -0.78 | -4.46 | -2.79 | 2.19 | 4.00 | -4.60 | -1.14 | -5.05 | 2.10 | 12.80 | -0.77 | -9.03 | -4.07 | 63.68 | | | | | |
| 11.50 | 9.16 | -0.83 | -4.36 | -3.03 | 2.16 | 3.99 | -3.87 | 0.12 | -7.40 | 1.73 | 12.54 | -0.90 | -8.90 | -6.54 | 80.44 | | | | | |
| 12.00 | 8.65 | -0.89 | -4.13 | -3.21 | 2.10 | 4.68 | -2.37 | -1.10 | -5.91 | 1.63 | 10.54 | -0.95 | -7.21 | -5.90 | 50.40 | | | | | |

LNZ: low-noise PHEMT amplifier stage
 AMP1: GaAs FET amplifier stage
 AMP2: Complete amplifier (LNZ and AMP1 cascaded)

that a typical transverter noise figure is 3 or 4 dB, it is pretty obvious that preamps with gains of 10 dB are more useful for winning noise-figure contests than for setting system noise figures. To get that ultra-low 0.4-dB system noise figure, you really need 20 dB of preamplifier gain, rather than 10 dB. To achieve that, you need a multistage amplifier. But the dynamic range of high-gain preamps tends to be less than that of lower-gain designs. Fortunately, the degradation in dynamic range is often acceptable as you go up in frequency, since signals can be separated using antenna directivity. Of course, if you already have a good system or just want to evaluate a device, single-stage pre-amps make more sense.

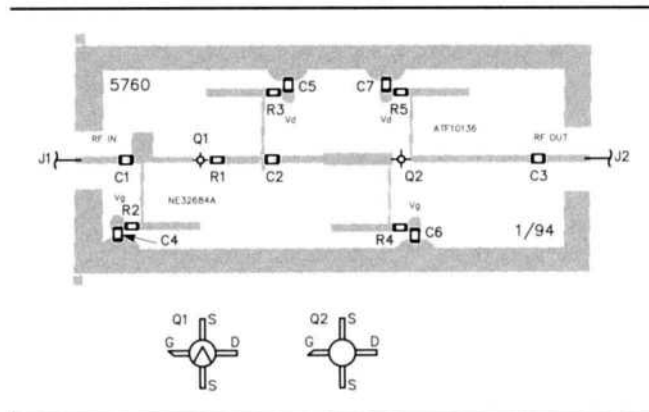
The gain-versus-frequency curve is also important. Ideally, your design shouldn't have a low-frequency gain peak; it should have more of a band-pass response, centered at the design frequency. This is particularly important if you are using broadband antennas near a cellular phone site. If you need to fix a system with this problem, you might consider using a horn feed, which will act as a waveguide below cutoff for low-frequency signals.

The Actual Design

The RF circuit of the preamp is shown in Fig 1, and the bias circuit is in Fig 2. Based on my success with the device at 10 GHz, I chose the NEC 32684A PHEMT for the low-noise stage. For the second stage I use a Hewlett Packard ATF 10135 GaAs FET. While the GaAs FET has a higher noise figure than the PHEMT, it is much easier to match for both noise figure and return loss. It is also significantly cheaper, while only having a slightly worse noise figure. Using the PHEMT as a second stage would lower the noise figure by just a few hundredths of a dB. If you had another PHEMT device to use, I'd recommend you build a second two-stage preamp and select the best one. It's not unusual for seemingly identical preamps to vary by a tenth of a dB in noise figure.

The circuit is etched on 15-mil 5880 Duroid board. I first built the GaAs FET stage on this board to see if there was any significant reduction in board loss compared to a design on 31-mil stock—there wasn't. And 15-mil board

Fig 5—Parts-placement diagram for the 5.7-GHz GaAs FET preamplifier.



allows shorter-length source leads when through-board mounting the transistors—and less consequent inductance. This is an important consideration for the PHEMT input stage, since this device has a lot of gain, even through Ku band.

While it made the design a bit more difficult, I wanted the flexibility of using either stage by itself. With calculated input and output return losses of at least 19 dB, or a 1.24:1 SWR, there shouldn't be any difficulty in using the stages of this design separately. However, if you use something other than SMA transitions, such as an N-to-microstrip transition, you may want to modify the matching circuitry slightly. It was designed for the SMA transition shown in Fig 3. The gain of the PHEMT stage is undoubtedly a little lower than what could be obtained if the stages were designed as a single unit, since some resistive loading is needed to get a stable single-stage design.

Construction

I advise reading about Al Ward's microwave preamps before building this one. You can find his article in either the May 1989 *QST* or the 1992 or 1993 *ARRL Handbook*. The transistor mounting is almost identical, though I decided to mark where to cut the slots. You want your slots to just touch the outside edges of the tiny PC board pads marking the slots. Ideally, you should get the long-lead versions of the devices. Otherwise, I'd recommend you use thin copper foil to connect the pads marking the slots to the ground plane, then solder the tran-

sistor to the copper foil. The foil should be 20 mils wide. I used EMI shielding tape with the adhesive removed. The tape is only 1 mil thick.

As with all my preamps, the bias circuitry is built over the ground plane side of the board. I prefer an active bias circuit, such as that shown in Fig 2, but you can use any circuit that doesn't overstress the PHEMT. According to the data sheet, the absolute maximum V_{ds} is 4 V, while the absolute maximum V_{gs} is -3 V. It is a good idea to test the negative voltage generator before connecting a bias supply to a microwave FET. Since the negative voltage is needed to turn the FET off, some designs can stress the transistor with too much drain current if there is a problem with the bias supply.

The file LNX5670.ZIP contains a Postscript file that can be used to print the circuit-board pattern, as well as the Microwave Harmonica circuit model description. This file can be downloaded from the ARRL BBS (203-666-0578), or via Internet from ftp.cs.buffalo.edu in the /pub/ham-radio directory.

Erratum

In the July column, Fig 2, the PC board pattern for the 13-cm converter microwave circuitry, was reproduced at the wrong size. A corrected copy of the pattern can be obtained from ARRL. Send a self-addressed, stamped envelope to: Technical Department, ARRL, 225 Main Street, Newington, CT 06111. Request the Mode-S template from the July, 1994 *QEX*. □