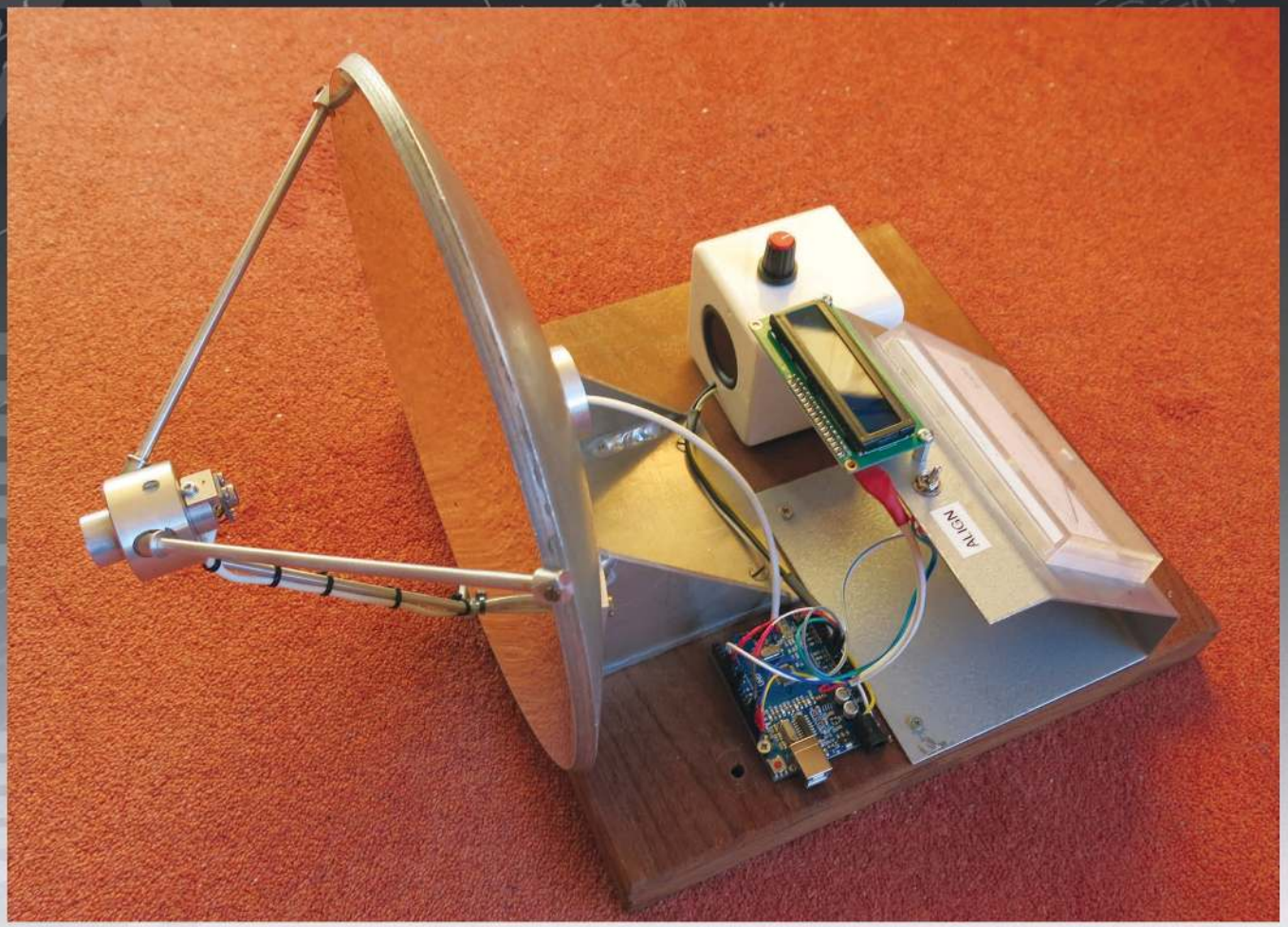
QEX

March/April 2022

www.arrl.org

A Forum for Communications Experimenters

Issue No. 331



G8AGN builds and operates simple 30 THz equipment.

The EVENT HORIZON OF DX

TS-990S

Dual TFT Display & Dual Receiver HF/50 MHz Transceiver

The main receiver has an IP3 in the +40 dB class, and the sub receiver is the already famous TS-590S receiver. Capable of receiving two signals at once, on different bands. 7-inch and 3.5-inch color TFT displays allow displaying of independent contents. Simplification of complex operations at a glance. Make no mistake, this is not a toy. Finally a serious tool is available for getting the very most from your hobby, of course it's a Kenwood.

- Covers the HF and 50 MHz bands.
- High-speed automatic antenna tuner.
- USB, Serial and LAN ports.
- Various PC applications (free software): ARCP-990 enabling PC control, ARHP-990 enabling remote control, and ARUA-10 USB audio driver.
- Clean 5 to 200 W transmit power through the 50 V FET final unit.
- Built-in RTTY and PSK.
- Three Analog Devices 32-bit floating-point arithmetic DSPs.
- DVI output for display by an external monitor (main screen display only).

KENWOOD

Customer Support: (310) 639-4200
Fax: (310) 537-8235

www.kenwood.com/usa

ISO9001 Registered

ADS#05421

QEX (ISSN: 0886-8093) is published bimonthly in January, March, May, July, September, and November by the American Radio Relay League, 225 Main St., Newington, CT 06111-1400. Periodicals postage paid at Hartford, CT and at additional mailing offices.

POSTMASTER: Send address changes to: QEX, 225 Main St., Newington, CT 06111-1400
Issue No. 331

Publisher
American Radio Relay League

Kazimierz "Kai" Siwiak, KE4PT
Editor

Lori Weinberg, KB1EIB
Assistant Editor

Scotty Cowling, WA2DFI
Ray Mack, W5IFS
Contributing Editors

Production Department

Becky R. Schoenfeld, W1BXY
Publications Manager

Michelle Bloom, WB1ENT
Production Supervisor

David Pingree, N1NAS
Senior Technical Illustrator

Brian Washing
Technical Illustrator

Advertising Information

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

Circulation Department

Cathy Stepina
QEX Circulation

Offices

225 Main St., Newington, CT 06111-1400 USA
Telephone: 860-594-0200
Fax: 860-594-0259 (24-hour direct line)
Email: **qex@arrl.org**

Subscription rate for 6 print issues:

In the US: $29

US by First Class Mail: $40;

International and Canada by Airmail: $35

ARRL members receive the digital edition of *QEX* as a member benefit.

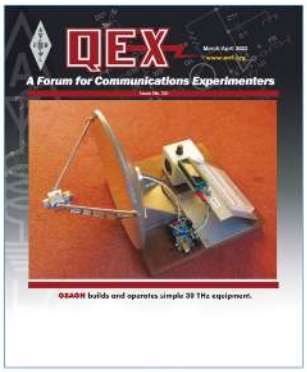
In order to ensure prompt delivery, we ask that you periodically check the address information on your mailing label. If you find any inaccuracies, please contact the Circulation Department immediately. Thank you for your assistance.

Copyright © 2022 by the American Radio Relay League Inc. For permission to quote or reprint material from *QEX* or any ARRL publication, send a written request including the issue date (or book title), article title, page numbers, and a description of where and how you intend to use the reprinted material. Send the request to **permission@arrl.org**.

March/April 2022

About the Cover

Barry Chambers, G8AGN, explores the 30 THz band with his simple 30 THz receiver using a Melexis 90614 sensor, and thermally based transmissions. The sensor data, in the form of the sensor ambient temperature and the object temperature, is read using an Arduino Uno with the Adafruit 90614 sensor library. The received infrared signal strength provides an audio output, at a user-chosen constant frequency, when the measured apparent temperature of the distant heat source is greater than the sensor ambient temperature. The transmitter is modulated by switching it on and off as very slow Morse code CW (QRSS). This system has set the UK distance record for 30 THz at 65 m.



In This Issue

Index of Advertisers

The American Radio Relay League

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

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

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

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

Membership inquiries and general correspondence should be addressed to the administrative headquarters:

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

Officers

President: Rick Roderick, K5UR
P.O. Box 1463, Little Rock, AR 72203

The purpose of *QEX* is to:

1) provide a medium for the exchange of ideas and information among Amateur Radio experimenters,

2) document advanced technical work in the Amateur Radio field, and

3) support efforts to advance the state of the Amateur Radio art.

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

Both theoretical and practical technical articles are welcomed. Manuscripts should be submitted in word-processor format, if possible. We can redraw any figures as long as their content is clear. Photos should be glossy, color or black-and-white prints of at least the size they are to appear in *QEX* or high-resolution digital images (300 dots per inch or higher at the printed size). Further information for authors can be found on the Web at **www.arrl.org/qex/** or by e-mail to **qex@arrl.org**.

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

Kazimierz "Kai" Siwiak, KE4PT

Perspectives

Re-cycling, Adapting, Adopting

The January / February 2022 *Perspectives* on "Re-cycling Electronics" drew several responses, including one from Bob Simmons, WB6EYV, who also laments (we paraphrase and quote his remarks), that "the days of salvaging televisions is gone forever... and that extracting resistors and capacitors from phenolic terminal strips and Bakelite tube sockets was about as genteel a process as a wisdom tooth extraction." Bob further notes that "rather than trying to salvage parts and save mere pennies, we might better encourage and cultivate new alternative technologies, and skills that would better benefit home brewers. At one time, the only skills a good ham needed were the ability to copy [Morse] code, and to use a soldering iron. Everything else involved skills that you could find in a carpenter's shop."

Bob gave examples of his projects where he employed materials like Kapton®, Delrin® and older standbys like acrylic plastic (PLEXIGLAS®), using modern techniques like computer controlled laser cutting technologies. We can add 3D printing to the list of modern techniques, and the widespread application of single board micro-controllers, and writing of micro-controller code, to the list of modern technologies and skills.

We asked, "what can today's home brewer recycle into a ham project?" It's now about adapting and adopting new technologies, and new skills.

In This Issue:

- Rick Muething, KN6KB, Tom Lafleur, KA6IQA, and Tom Whiteside, N5TW, describe IONOS, an ionospheric simulator for ARQ protocols.
- Barry Chambers, G8AGN, builds and operates simple 30 THz equipment.
- Arlen Young, K6KZM, builds a loop array for receiving low-band signals.
- Eric Nichols, KL7AJ, in his Essay Series, investigates ac electronic behavior.
- Brian Callahan, AD2BA, repurposes portions of a transmitted image to carry other information.

Writing for *QEX*

Please continue to send in full-length *QEX* articles, or share a **Technical Note** of several hundred words in length plus a figure or two. *QEX* is edited by Kazimierz "Kai" Siwiak, KE4PT, (**ksiwiak@arrl.org**) and is published bimonthly. *QEX* is a forum for the free exchange of ideas among communications experimenters. All members can access digital editions of all four ARRL magazines: *QST*, *On the Air*, *QEX*, and *NCJ* as a member benefit. The *QEX printed edition* is available at an annual subscription rate (6 issues per year) for members and non-members, see **www.arrl.org/qex**.

Would you like to write for *QEX*? We pay $50 per published page for full articles and *QEX* Technical Notes. Get more information and an Author Guide at **www.arrl.org/qex-author-guide**. If you prefer postal mail, send a business-size self-addressed, stamped (US postage) envelope to: *QEX* Author Guide, c/o Maty Weinberg, ARRL, 225 Main St., Newington, CT 06111.

Very kindest regards,
Kazimierz "Kai" Siwiak, KE4PT
QEX Editor

Barry Chambers, G8AGN

5 The Ridge, Sheffield, South Yorks, S10 4LL, England; **b.chambers@sheffield.ac.uk**

30 THz – It's Radio, But Not As You Know It

There is a lot to explore using simple to build and operate 30 THz equipment.

Introduction

Radio amateurs have always been at the forefront of exploring and exploiting hitherto regarded "as useless" frequency bands and today is no exception. For some years now, radio amateurs have been exploring frequencies in the range 100 to 1000 GHz (0.1 to 1 THz) and in the band around 474 THz (light waves). To put these frequencies into perspective, 1 THz corresponds to a wavelength of 0.3 mm and when dealing with light waves, the wavelengths involved are around 600 nm (0.0006 mm). Through the air RF communication at frequencies above 100 GHz is difficult for radio amateurs using conventional techniques because of the high cost and low availability of suitable components, low transmitter powers and the fact that the atmosphere is becoming more opaque with increasing frequency, primarily because of water vapor and oxygen absorption. Operating at light wave frequencies, however, is considerably easier because of the ready availability of inexpensive high-power red LEDs, photo detectors and high gain antenna systems based on Fresnel lenses; furthermore, the atmosphere is fairly transparent to light waves and hence contacts over distances of several hundred miles have already been achieved by a number of radio amateurs.

The next challenge for radio amateurs is that of exploring the frequency range between, say, 1 THz and 470 THz. Over much of this range, the atmosphere is opaque, but there are "windows" in which the transmission losses are much lower. Of particular interest to radio amateurs is the so-called Long Wavelength Infrared (LWIR) band, corresponding to frequencies around 30 THz and wavelengths around 0.01 mm. In this band, the atmosphere is transparent and this fact has long been appreciated by the military and security agencies but the equipment that they operate is either not available to the experimenter or is extremely expensive. In recent YouTube videos **[1]**, **[2]**, however, Andrew Anderson, VK3CV / WQ1S, was the first amateur to demonstrate that it is possible to operate at these frequencies by using readily available and inexpensive sensors such as those found in the ubiquitous remote reading infrared thermometers. These sensors are made by several manufacturers but one in particular, the Melexis MLX90614 **[3]**, will be considered here since it is readily available from eBay and is already mounted on a small PCB which also contains the additional components needed for a simple I2C interface to a microcontroller such as an Arduino. To make life even easier, Adafruit have already developed an Arduino library for the sensor, which may be downloaded from their web site **[4]** or by using the Arduino IDE Library Manager.

Before describing my equipment for use on the 30 THz band, it is necessary to give a brief overview of the way in which electromagnetic waves interact with objects at these frequencies. The starting point is the fact that every natural object in the Universe radiates electromagnetic waves because its associated temperature is above absolute zero (0 K or –273.15 °C). The coldest "object" is left-over radiation from the "Big Bang" and is at a temperature of about 2.7 K. The human body, at about 37 °C (310 K), radiates electromagnetic waves whose frequency follows the Black Body curve, peaking at around 32 THz (a wavelength of about 9.3 µm). The Sun, which has a surface temperature of roughly 6000 K, emits electromagnetic waves over a very wide range of wavelengths, but those lying between about 400 nm and 700 nm are perceived by our eye-brain as the visible color spectrum ranging from purple to red.

To a first approximation, the way in which real objects radiate electromagnetic energy is described by Planck's Law. This allows us to estimate the strength of the radiated signal from an object at a given temperature, as a function of wavelength. Typical radiation curves are shown in **Figure 1**. The wavelength range in the figure was chosen to match that of the MLX90614 sensor.

The temperatures used in computing the curves shown in **Figure 1** are typical of the human body, the boiling point of

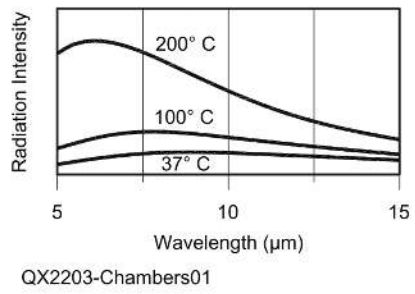


Figure 1 — Black-Body radiation characteristics for an object at various temperatures. Vertical axis denotes radiation intensity.

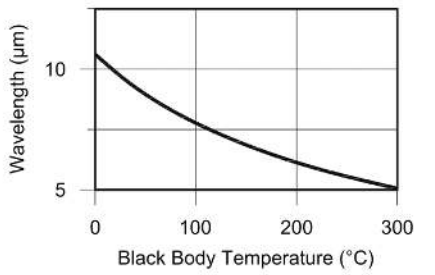


Figure 2 — Wien's Law shows the peak radiation wavelength vs. object temperature.

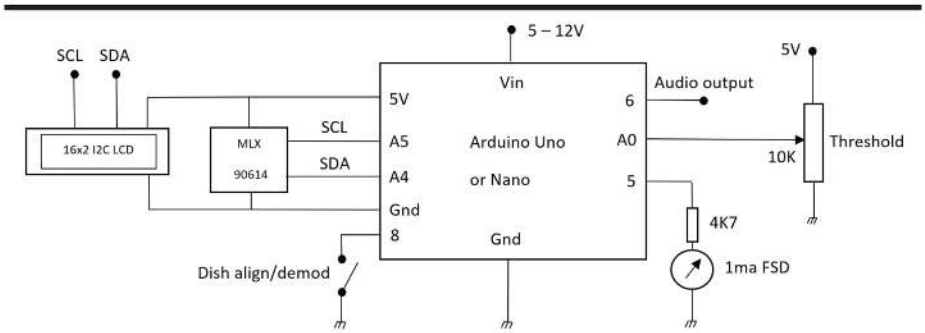


Figure 3 — Schematic of 30 THz receiver.

water and that of a typical domestic steam iron. It can be seen that an object radiates more energy the higher its temperature, the energy is radiated over a wide range of wavelengths (or frequencies) and the position of peak emission changes with object temperature. The relationship between an object's temperature and the wavelength at which maximum radiation of energy occurs is called Wien's Law and this is shown in **Figure 2** for part of the detection temperature range covered by the MLX90614 sensor (5.5 – 14 µm). Hence it can be seen from **Figure 2**, why the Melexis 90614 sensor is suitable for measuring the temperature of the human body (37 °C equates to a radiation peak at about 9.3 µm).

Strictly speaking, the radiation curves shown in **Figure 1** only apply to a fictitious object known as a Black Body. This is an object which is in thermal equilibrium in the sense that it acts both as a perfect absorber and a perfect emitter of energy over the entire frequency spectrum; hence its temperature remains constant. Real objects are often referred to as Gray Bodies since not only do they not absorb all the energy incident upon them — some may be reflected or scattered — but the amount of energy emitted does not necessarily equal the amount absorbed; hence their temperature may change with time and circumstances. A night storage heater or a cup of coffee are good examples.

Thus, a Black Body is a theoretically perfect entity that is used as a standard against which to compare the properties of real objects and materials. It is analogous to the idea of comparing the gain of a real antenna with that of an isotropic radiator, which also does not exist in practice. By comparing the properties of a Gray Body with those of a Black Body it is possible to determine which materials will be suitable for particular purposes in our 30 THz equipment. For example, a curved highly polished metallic surface should act as a good parabolic reflector antenna at 30 THz whereas a painted metallic surface would not. Conversely, a heated black anodized aluminum surface — like a transistor heat sink — should act as a reasonable emitter at 30 THz and so might make a good choice for a simple test source, as would a human hand with its rough textured skin.

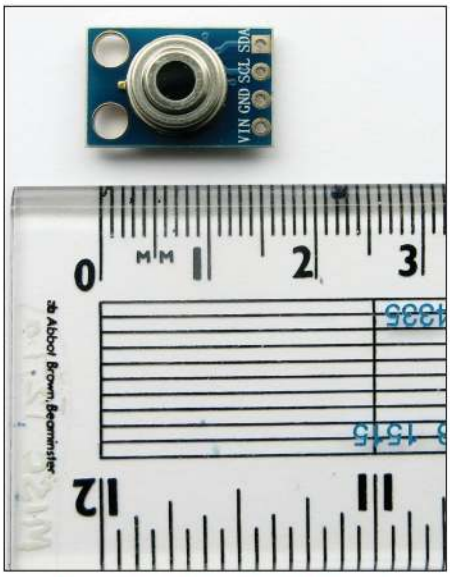


Figure 4 — Melexis 90614 sensor.

A simple receiver for 30 THz

The circuit schematic of my receivers is shown in **Figure 3**. At its heart is a Melexis 90614 sensor, which is available on eBay for about US$10 and is supplied on a small PCB together with the few additional components needed to implement an I2C interface. A close up of this PCB is shown in **Figure 4**. The sensor data, in the form of two temperature values, the sensor ambient temperature and the object temperature, may be read using an Arduino Uno or Nano and the Adafruit 90614 sensor library. When the measured object temperature is greater than the sensor ambient temperature, the difference is considered to be a measure of the received infrared signal strength. This signal level can be interpreted in several ways. For example, the temperature difference can be displayed as a number in °C, as a meter deflection or converted into an audio tone whose pitch increases with increasing signal level. By these means, the sensor when mounted at the focus of a parabolic dish, can be pointed accurately in the direction of the infrared transmitter. Once the receiving antenna has been aligned for maximum received signal, the receiver can be switched from the **dish align** mode to the **demodulate** mode; in this case it merely provides an audio output, at a user-chosen constant frequency, but only when the measured apparent temperature of the distant heat source — for example, the transmitter — is greater than the sensor ambient temperature. Hence, if the transmitter is then modulated by switching it **ON** and **OFF** in the appropriate way, the result at the receiver will be an audio tone that sounds like normal CW. In practice, because of the slow response of the sensor, very slow Morse code CW (QRSS) rather than normal speed CW may have to be transmitted.

As shown in **Figure 3**, the receiver audio tone output is taken from Arduino pin D6 and Gnd, and can be fed either to a small audio amplifier and speaker or just to a speaker via a 470 Ω potentiometer (not shown) configured as a variable resistor. An SPST toggle switch is connected between Arduino pin D8 and Gnd. When the switch is open, the receiver is in **dish align** mode and in **demodulate** mode when it is closed. A signal strength meter is driven by the voltage output at Arduino pin D5. The output range is nominally between 0 and 5 V, the latter value indicating a very strong received signal — that is, a very

high measured source temperature. I had to drive a 0 to 1 mA FSD meter, so this required an additional series resistance of 4.7 kΩ. Meters with other FSD values can be accommodated by an appropriate choice of series resistance but the current drawn from Arduino pin D5 should be limited to no more than about 10 mA. If actual values of the measured source and sensor ambient temperatures are required, these can be displayed using an optional 16×2 I2C LCD display, which is updated once every second when the receiver is in the **dish align** mode.

As mentioned earlier, the Melexis 90614 sensor is used in the receiver to measure the sensor temperature T_{sensor} and the apparent heat source (distant transmitter) temperature T_{source}. Then a valid received signal requires that $T_{source} - T_{sensor} > 0$. In practice, the receiver field of view is large enough that the source is seen against a background which also emits an infrared signal since it is at a temperature $T_{background}$. In my early testing outdoors, the outside temperature and background were typically around or below 0 °C, but my sensor temperature was above this and so the detection criterion given above was valid. A problem occurs, however, when $T_{background} > T_{sensor}$ since this results in a valid signal being received even when the source is not switched on. This can be avoided by using the detection criterion $T_{source} - T_{sensor} > X$ where X can be thought of a variable threshold which can be adjusted at will in the same way as the squelch control on a FM receiver. In practice, a value for X can be set using a potentiometer and read using ADC input A0 on the receiver Arduino, as shown in **Figure 3**. Hence, when the source is not switched on, the value of X is increased until the receiver LCD display shows "NO SIGNAL". Since $X = T_{background} - T_{sensor}$ it would be possible to set the value of X automatically by reading these values during the spaces between **DITs** and **DAHs** when the transmitter is sending QRSS, but the manual "squelch" control is easier to implement.

My first 30 THz receiver used a 10-inch 0.46 *f/D* parabolic reflector antenna with the Melexis sensor mounted at its focus, as shown in **Figures 5** and **6**. The reflector had originally been a searchlight mirror and is made of cast aluminum whose surface has been silver plated. A perfectly acceptable alternative would be to use a normal aluminum prime focus dish, which has had any paint layer removed and the metal surface polished using a propriety chrome cleaner, such as is widely available from hardware and automobile accessory stores. Such a reflector has been used in my second receiver, shown in **Figure 7**. The Melexis sensor has a 90° field of view and so the ideal dish *f/D* would be around 0.5 but this is not critical. My second receiver uses a focal plane dish with *f/D* = 0.25, and this works well. It is possible to obtain other reflectors such as concave shaving mirrors, but these are not suitable because the reflecting surface is at the back of the glass and glass is not transparent at 30 THz. Similarly, common glass lenses are not viable at 30 THz, but others using certain plastics such as polyethylene or unusual materials such as germanium are available but the latter material is expensive and toxic.

The receiver can be tested for correct operation merely by switching to the **dish align** mode and placing a warm object — even a finger — close to the sensor when an audio tone should be heard. The hotter the object is, the higher the audio tone pitch should be. Other convenient test sources

Figure 5 — Melexis sensor mounted at the focus of a 0.46 f/D parabolic reflector.

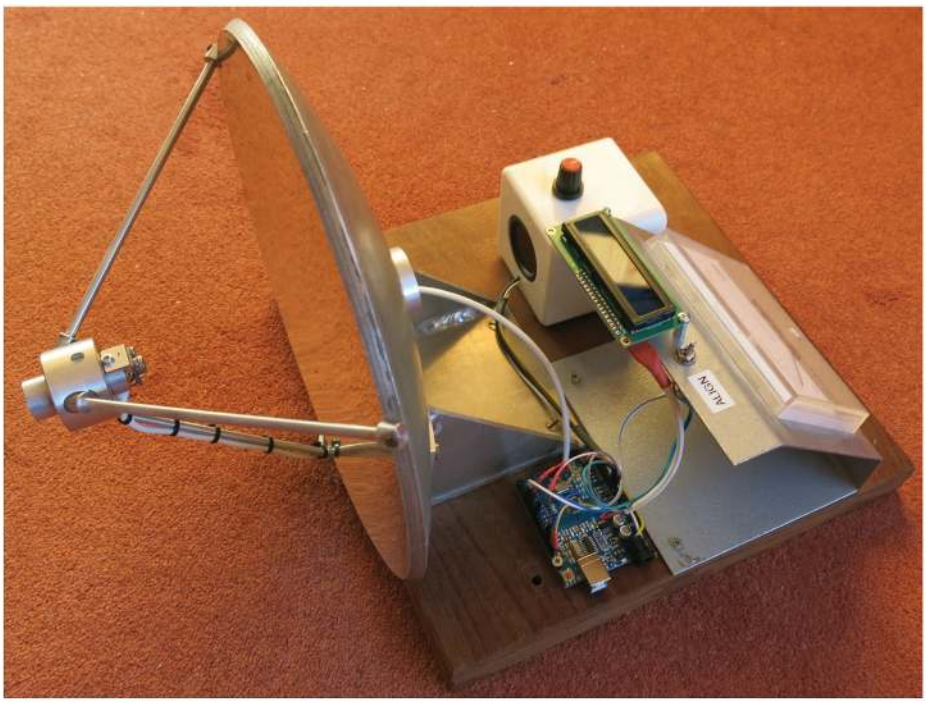


Figure 6 — The author's first 30 THz receiver.

(A) (B)

Figure 7 — (A) Focal plane dish used in the author's second 30 THz receiver. (B) the author's second 30 THz receiver.

are a domestic hot water radiator or a kettle containing recently boiled water.

A QRSS modulated 30 THz source

The Melexis 90614 receive sensor has a typical response time of around half a second. Furthermore, since it is impractical to apply fast modulation rates to inexpensive high power large area heat sources such as are required for this application, it would appear that QRSS must be used. Before describing my QRSS transmitter in detail, however, some background discussion is required. After careful consideration, I settled on using a heat source (infra-red transmitter) in the form of a square, black painted aluminum plate, which is heated using an array of 50 W dissipation resistors mounted on its rear face. Three main questions then need to be answered. These are (a) how large should the plate be, (b) what temperature should the plate be heated to and (c) how is modulation to be applied?

The first two questions may be answered by consideration of the Stefan-Boltzmann Law, which states that the power radiated over the whole frequency spectrum from 0 to ∞ by an object at a temperature T (measured in K) and area A is given by $P = \sigma T^4$, where $\sigma = 5.67 \times 10^{-8}$ Wm^2K^{-4} is the Stefan-Boltzmann constant in MKS units. This relationship is rather similar to that used to calculate the effective radiated power of a RF transmitter, namely the product of the transmitter output power multiplied by the antenna gain. The difference is that in the RF case, the radiated power is normally only emitted over a carefully controlled bandwidth whereas in the thermal case the power is emitted (in theory at least) over an infinite bandwidth. In practice, as can be seen from the emission curves shown in **Figures 1 and 2**, the thermal radiation is band-limited and what really matters is how much of the emitted radiation is actually

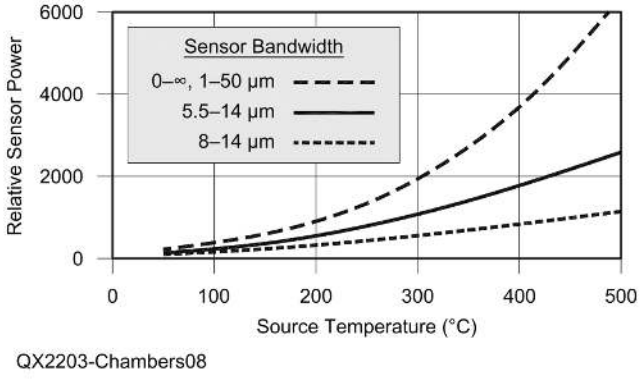


Figure 8 — Power collected by Melexis 90614 sensor for various source temperatures.

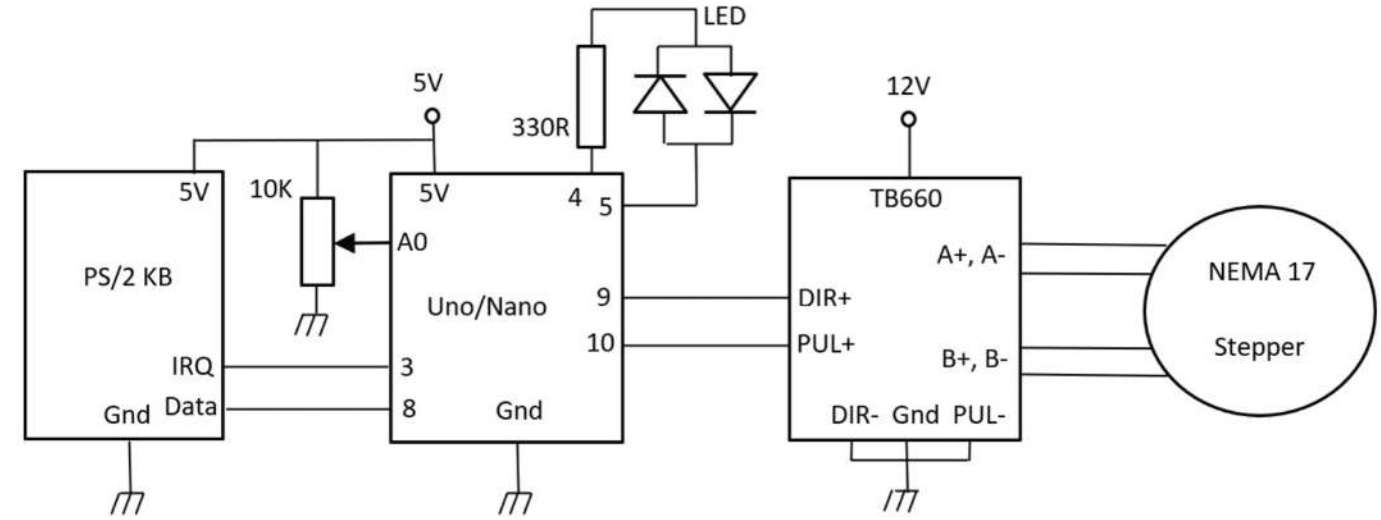


Figure 9 — Driver for rotating heated plate 30 THz source.

utilized by the receiver sensor, which in our case has an effective bandwidth between 5.5 and 14 µm. This question is answered by the data shown in **Figure 8** for various source temperatures. Two curves are shown for the sensor bandwidth since for long path lengths, it is expected that thermal radiation over the wavelength range from 5.5 to 8 µm will be attenuated by water absorption. In conclusion, therefore, **Figure 8** and the Stefan Boltzmann Law show that our heated plate source should be as hot and as large in area as possible.

Turning now to the question of modulation, it is already apparent that QRSS seems to be the only form of modulation that can be used. Because of thermal time constant considerations, this cannot be applied to the source by changing the temperature, so we are only left with applying some means of source shuttering. This could be done using a mechanically actuated system, such as that used in naval signaling lamps or in automobile rotating hazard warning lights, or even using a moveable mirror such as in a heliograph. Having considered all these systems, I finally decided on a shuttered system based on rotating the heated plate using an Arduino-controlled stepper motor. The circuit schematic of my system is shown in **Figure 9**. A QRSS message to be sent is entered into the modulator using a PS/2 keyboard and the Arduino coverts this into a stream of **DITs** and **DAHs** with the appropriate timings for the QRSS mode in use (QRSS3, 2 and 1.5 are currently available). The **DITs** and **DAHs** are then translated into appropriate heated plate rotation instructions for the NEMA 17

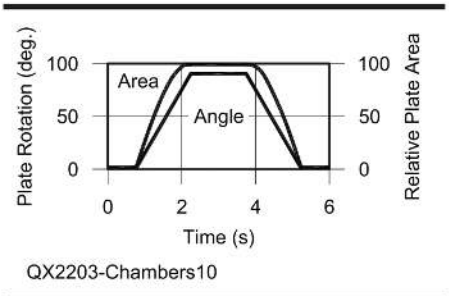


Figure 10 — Characteristics of QRSS3 Morse DIT time.

stepper motor, which is driven by a TB6600 interface module.

Figure 10 shows the characteristics of my current QRSS3 **DIT**. The latter consists of a 3 second **ON** time with a one-half **DIT** time (1.5 seconds) on either side. The latter is to ensure the correct one **DIT** time spacing between adjacent QRSS elements. The **ON** time is defined here by the length of time during which the visible plate area is greater than 70%. The lower curve shows the plate rotation angle against time and the upper curve shows the resulting area of the plate that is visible to the receiver. The latter is defined as 100sin(*plate_rotation_angle*). The corresponding QRSS3 **DAH** will have the same structure as that shown in **Figure 10**, but the time for which the plate is resting at the 90° position will be lengthened by an extra two **DIT** lengths (6 seconds). Thus, the total **DAH** time will be 12 seconds.

The number of times that a message is sent can be varied using a 10 kΩ potentiometer whose wiper is connected to the Arduino ADC input pin A0, and there is also the facility to transmit a small number

Figure 11 — Mechanical arrangement of the rotating plate 30 THz source.

of pre-stored messages for test purposes. A red-green LED is used as a modulator status indicator.

Figure 11 shows the general arrangement of the rotating heated plate source. It consists of a black painted aluminum square plate, which is heated by an array of 50 W dissipation resistors that are situated on its rear face. The plate is housed in a square tin-plate enclosure (a cake baking tin) with glass wool insulation between the latter and the plate. The plate is suspended from the enclosure by four very thin enameled copper wires which are attached at the plate corners.

This arrangement ensures minimum plate heat loss except from the radiating front face. The plate heating resistors are fed with a 15 V, 20 A dc supply which results in a typical plate temperature between 250 and 300 °C, depending on outdoor environmental conditions.

Conclusion

At the time of writing, the system described above has been used to set the UK distance record for 30 THz at 65 m. Equipment for 30 THz is simple to build and operate but there is still much to be explored, especially about propagation. It is anticipated that the distance records mentioned above by no means represent the limit for future contacts on this band.

The Arduino sketches that control the receiver and transmitter are available by contacting the author by e-mail: **b.chambers@sheffield.ac.uk**.

Barry Chambers, G8AGN, was licensed in 1965 and made his first microwave contact in 1968 on 3.4GHz. Since then, he has worked on all the microwave bands from 1.3 GHz to 134 GHz as well as on the 30 THz, UV, visible and IR nanowave bands. Most of his operating has been done portable. Before his retirement, Barry was Professor of Electromagnetic Engineering at the University of Sheffield in the UK and head of the Antennas Research Group. His research interests included microwave antennas and propagation, free-space microwave measurements and smart microwave materials and structures. He was elected Fellow of the IEEE for his pioneering contributions. Over the years, Barry has served as a member of the RSGB Microwave and Propagation Studies Committees and the UK Microwave Group committee. He has written a number of articles for the RSGB RadCom *magazine and is a frequent contributor to* Scatterpoint *the UK Microwave Group newsletter.*

References

[1] Equipment test #4 – YouTube; **https://www.youtube.com/watch?v=y3A1zQi8V5Q**.
[2] Full system demo – YouTube; **https://www.youtube.com/watch?v=6gJtzMLR6T0**.
[3] MLX90614 datasheet; **https://www.melexis.com/en/documents/documentation/datasheets/datasheet-mlx90614**.
[4] MLX90614 Library; **https://www.arduino.cc/reference/en/libraries/**

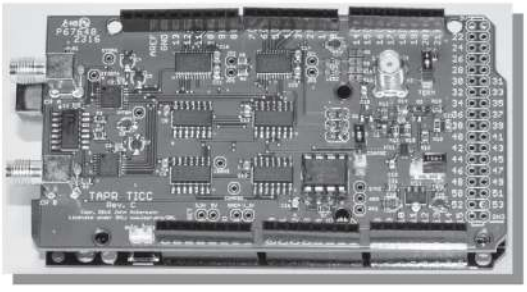
Announcement

Ulrich L. Rohde, N1UL, a regular *QEX* and *QST* author, is an Editor of the book, "Fundamentals of RF and Microwave Techniques and Technologies," by Editors Hans-Ludwig Hartnagel, Rïdiger Quay, Ulrich L. Rohde, and Matthias Rudolph, 2022, published by Springer International Publishing. The book provides state-of-the-art coverage of RF and microwave techniques and technologies, and covers topics that include transmission-line theory, semiconductor devices, active and passive microwave circuits and receiver systems, as well as antennas, noise and digital signal modulation schemes. This book targets students, teachers, scientists, and practicing design engineers who are interested in broadening their knowledge of RF and microwave electronic circuit design.

TAPR + WSPR

TAPR has 20M, 30M and 40M WSPR TX Shields for the Raspberry Pi. Set up your own HF WSPR beacon transmitter and monitor propagation from your station on the wsprnet.org web site. The TAPR WSPR shields turn virtually any Raspberry Pi computer board into a QRP beacon transmitter. Compatible with versions 1, 2, 3 and even the Raspberry Pi Zero! Choose a band or three and join in the fun!

TAPR is a non-profit amateur radio organization that develops new communications technology, provides useful/affordable hardware, and promotes the advancement of the amateur art through publications, meetings, and standards. Membership includes an e-subscription to the TAPR Packet Status Register quarterly newsletter, which provides up-to-date news and user/technical information. Annual membership costs $30 worldwide. Visit **www.tapr.org** for more information.



TICC

The **TICC** is a two channel time-stamping counter that can time events with 60 picosecond resolution. Think of the best stopwatch you've ever seen and make it a hundred million times better, and you can imagine how the TICC might be used. It can output the timestamps from each channel directly, or it can operate as a time interval counter started by a signal on one channel and stopped by a signal on the other. The TICC works with an Arduino Mega 2560 processor board and open source software. It is currently available from TAPR as an assembled and tested board with Arduino processor board and software included.

TAPR

1 Glen Ave., Wolcott, CT 06716-1442
Office: (972) 413-8277 • e-mail: taproffice@tapr.org
Internet: www.tapr.org • Non-Profit Research and Development Corporation

Rick Muething, KN6KB
6143 Anchor Ln., Rockledge, FL 32955; **rmuething@cfl.rr.com**

Tom Lafleur, KA6IQA
Box 3029, Rancho Santa Fe, CA 92067; **tom@lafleur.us**

Tom Whiteside, N5TW
228 Wind Ridge Cv., Georgetown, TX 78628; **tomw@ecpi.com**

IONOS Simulator: An Open Source Ionospheric Simulator

A low cost stand-alone open source ionospheric simulator for ARQ protocols.

This article covers the detailed design and implementation of a single chip microprocessor Ionospheric Simulator based on the popular Watterson model. This article also includes several of the results using the simulator to compare popular HF and VHF/UHF digital ARQ modes.

The project was begun in December 2019 as both a learning tool and proof of concept for a fully open-sourced project to implement a high-quality easy-to-use simulator based on the Watterson model. To make this successful we had to develop hardware and software that provided not only the required DSP manipulation of the audio but also provided a simple-to-learn user interface. A big factor in the successful development was the use of the Teensy Audio library — originally targeted primarily for music applications. We validated the simulator through extensive testing of several HF and VHF digital protocols and compared the simulated results with other Waterson model simulators. This article reviews the approaches taken for the system design, software, hardware, and application of the simulator and presents some of the more important simulation results. Since this is a fully open source project, we have provided all the design details, documentation, and source code on a GitHub site.

The Need for an Ionospheric Simulator

As hams we understand that radio propagation is heavily influenced by the earth's ionosphere. This ionized region of the upper atmosphere allows single or multiple hop communications through continuously-changing paths. In 1970 Watterson, Juroshek, and Bensema presented a landmark paper **[1]** in the IEEE Transaction on Communication Technology. They described a mathematical process (model) that could fairly accurately model and simulate radio propagation over narrow band HF (< 15 kHz) channels. This and follow-on papers by others, and the CCIR/ITU **[2]** defined a set of standardized "representative test channels" that would allow computer software or hardware simulators to statistically model HF radio propagation. This approach has been successful and simulators (hardware and software) built using the Watterson model have proved to be invaluable in the development of high-performance HF and VHF/UHF protocols for both military and ham radio use.

System Design

We wanted to leverage our experience with earlier projects with the higher performance Teensy microprocessors and the use of an earlier basic simulator described in a 1999 DCC paper **[3]** by Johan B. Forrer, KC7WW.

The approach taken was to use the high performance Teensy 4.0 microprocessor **[4]**. All programming and debugging were done using the simple and free Arduino Integrated Development Environment (IDE). We made good use of the Teensy Audio Library **[5]** that processes 16-bit audio (from a stereo sound card chip) at 44.1 kHz sample rate through various audio functions. Many of these library components were initially targeted to musical applications but we required only basic DSP functions such as multiplication, mixing, FFT, FIR filters, sine wave and random generators and rms and p-p measurement tools. A total of 35 Teensy Audio components were used connected by 38 virtual connections as shown in the detailed block diagram in **Figure 1**. This model is detailed and formalized in the C++ source code for the simulator.

We wanted the simulator to be self-contained and OS independent with its own basic user interface. This meant it would have to include at least basic parameter input and display capability. We also included an alternate serial USB interface to allow automation of longer more complex testing. The simulator contains a number of filters such as basic low-pass, band-pass and high-pass along with a few more complex filters for the generation of the imaginary Q components (Hilbert Filter) and the specialized very low frequency Gaussian filters needed for the Watterson model. All filters were designed with free and available on-line tools **[6, 7]**. Another requirement was the ability to use a single simulator to easily model both paths of an ARQ half-duplex connection without any external mixers or switching combiners. This was solved using the stereo input and output capability of Teensy Audio codex assigning one mono channel to each side of the connection. Two simulators would be required to model a full duplex connection. Finally, as shown in **Figure 1**, we included a second path through the Upmixer, FIR HP and Downmixer to

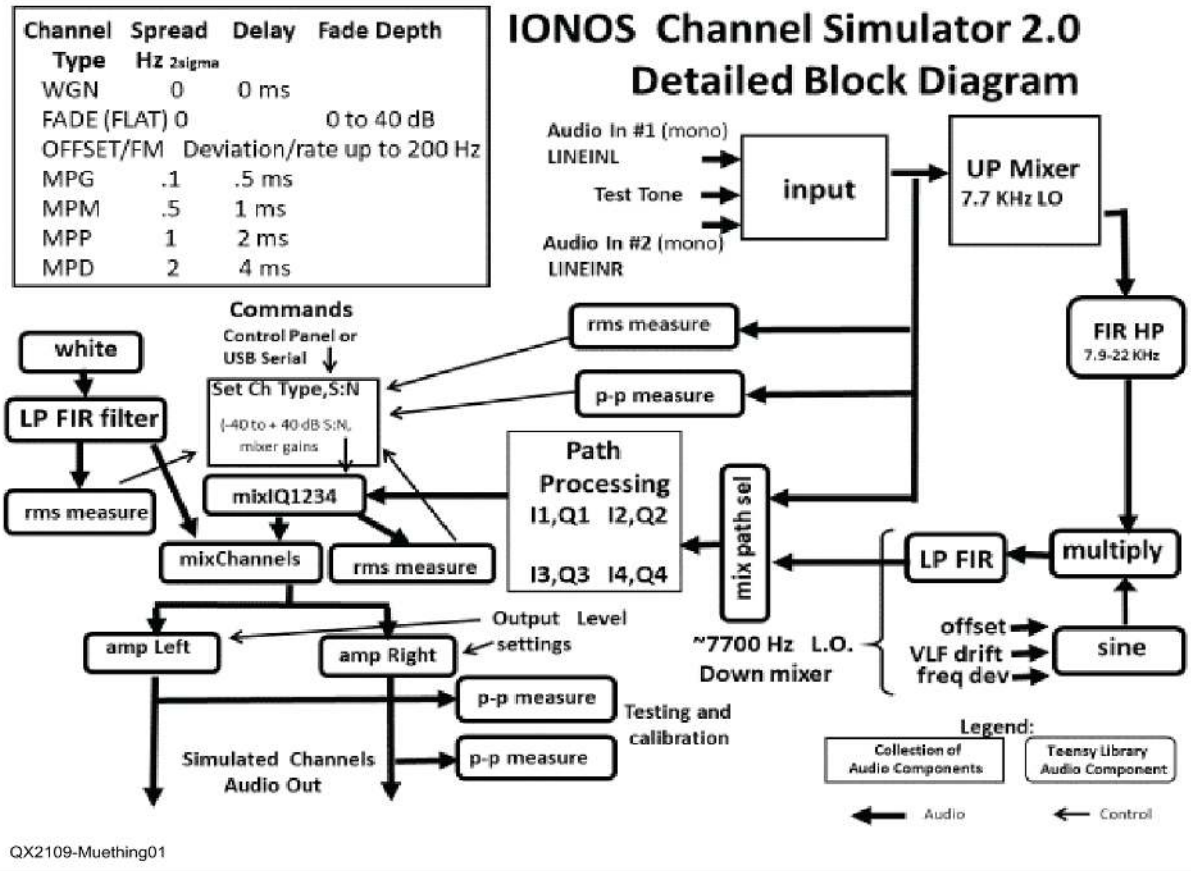


Figure 1 — Detailed block diagram of the IONOS simulator.

allow simulating tuning offset and FM drift important in testing some protocols.

In the upper left corner of **Figure 1** is a summary of the primary operating modes and channels of the simulator. The CCIR/ITU standardized the Doppler spread and delay values of representative channels MPG-MPP along with specific filter requirements to better allow comparison using different simulators.

The implementation of the Waterson model is expanded in the Path Processing Detail shown in **Figure 2**. This implements the modeling and simulation for two or four complex (I and Q) paths. In the Waterson model the variation of these I and Q paths are independent and randomized so this requires a total of four or eight individual paths to implement the two or four IQ pairs.

The basic audio (real component) of the modem output drives two delay elements delay I1 and delay I2 along with a Hilbert Filter, which shifts the real audio 90° to generate the Q (imaginary) component. The Hilbert filter drives two additional delay elements Q1, Q2 that delay the Q (imaginary) components. The I and Q delays are adjusted to offset the delay of the Hilbert filter and introduce the constant path delay defined by the channel to be modeled (e.g., 2 ms delay defined in the MPP channel shown in **Figure 1**). The CCIR/ITU guidelines also put specific requirements on the bandwidth, shape and roll-off of the low-pass filters that are used to generate the low frequency complex modulation of the audio I and Q paths. The final result output is generated by combining — with proper polarity — all the real components (total of 4 as shown in **Figure 2**). This audio, which is now modulated by a low frequency randomized complex vector, is what causes the frequency selective slow fading and peaking that we recognize on HF propagation. The imaginary components are not summed but discarded because only real audio is required by the modem. Additional details on how this important step is done and the details of the various filters used are included in the slide presentation on the GitHub page.

Finally, to accommodate high performance ARQ modes, we paid close attention to audio path alignment and processing delays to keep the overall audio transit time down to a maximum of less than 10 ms.

Hardware Design

The first proof of concept for this project was testing with a standalone Teensy 4.0 processor with the PJRC Teensy audio codex (shield) board rev D. (see **Figure 3**).

When we were satisfied with the concept, a rev 1 PCB was designed and fabricated that included an LCD display, 2 rotary encoders, the audio codex, some I2C devices, and a standalone power supply. This was used for the early development of the software. We then migrated to a version 2 board, for final development, optimizing a few items that were not needed to simplify the design.

The heart of the simulator is a Teensy 4.0 processor module from PJRC **[4]** that features an ARM Cortex-M7 processor with a floating-point unit, 1024 K RAM and 2048 K Flash EEPROM memory, running at 600 MHz, with a NXP iMXRT1062 processor chip. It's very fast and was selected to provide the horsepower needed to run the simulator code.

The board has two Bourns 12 mm incremental rotary encoders for user interface control, an NXP SGTL5000 16-bit stereo audio codec, support for a 2.2 in or 2.8 in (56 mm or 71 mm) TFT LCD display and op-amps for input buffers to isolate the codec. The op-amps are TI OPA2322, rail-to-rail I/O with ESD clamping diodes on the

inputs, and very low noise.

The encoders are first level de-bounced with two 10 kΩ resistor and a 0.01 µF capacitor on the encoder pins and a single 0.01 Ω capacitor on the switch pins. Second level de-bouncing is provided in software with the Encoder2 library package.

The SGTL5000 require both 3.3 V and 1.8 V power for proper operation. The 3.3 V is provide by the P3.3V output from the Teensy module low drop out regulator and filtered for clean power to the codec, an AP7313-18 LDO provides the 1.8 V from the 3.3 V source.

The LCD is a common 2.8 in (or 2.2 in) low cost 240x320 TFT device available from a number of suppliers under part number MSP2806 or MSP2807 with touch screen for the 2.8 in devices or MSP2202 for the 2.2 in. Connection to the Teensy is via an SPI interface. Both boards use an ILI9341

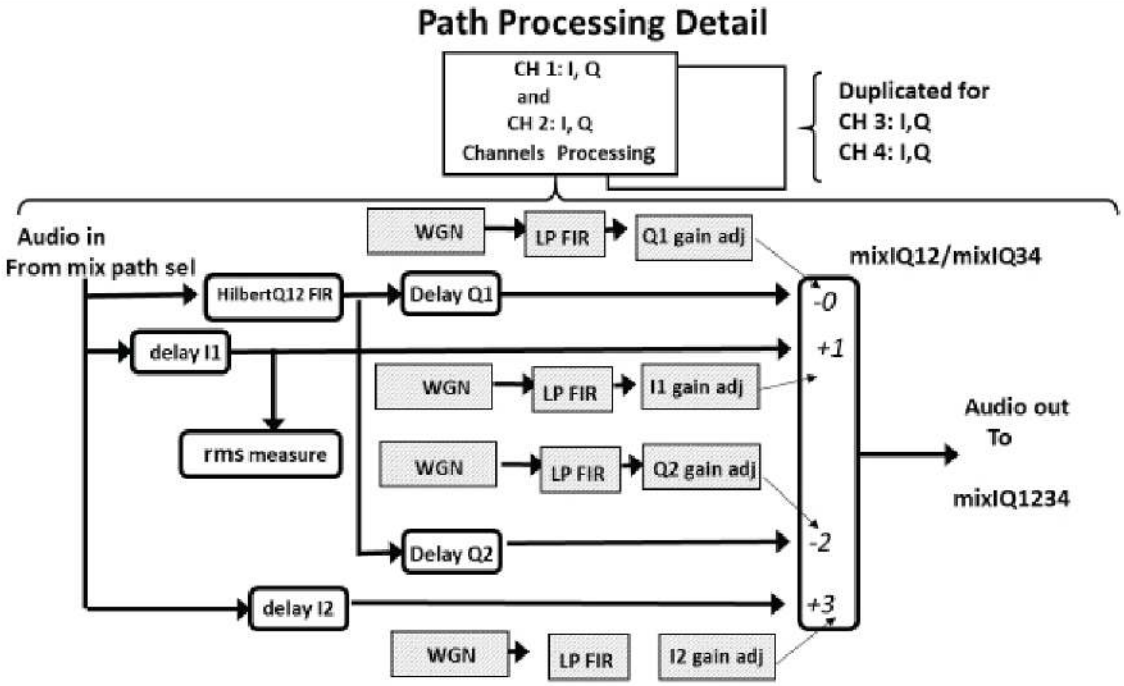


Figure 2 — Path processing detail for two and four I and Q paths.

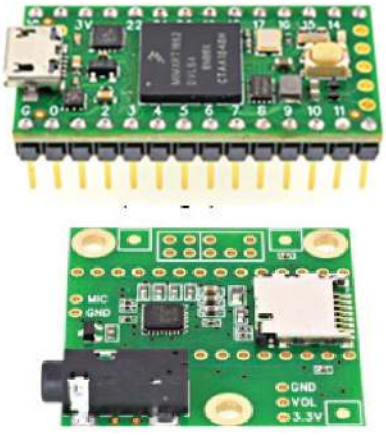


Figure 3 — Teensy 4.0 processor and audio shield.

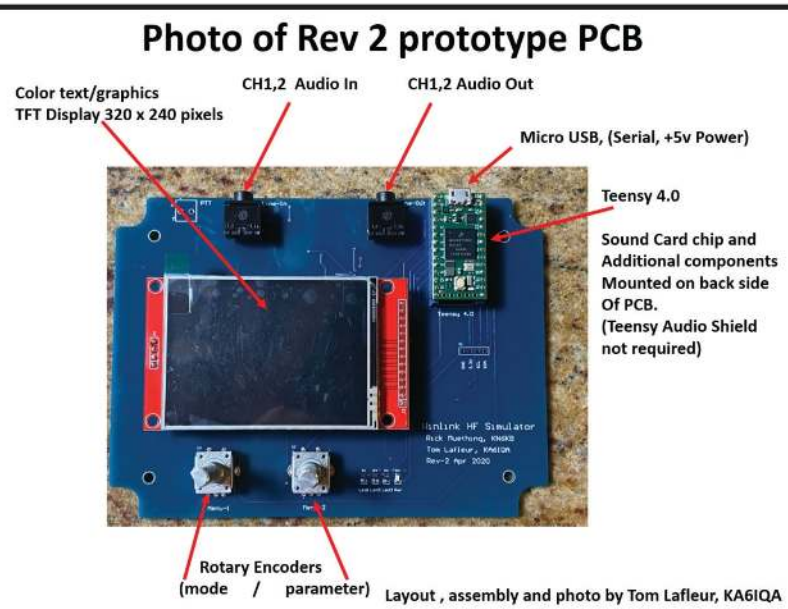


Figure 4 — Photo of Rev 2 prototype PCB.

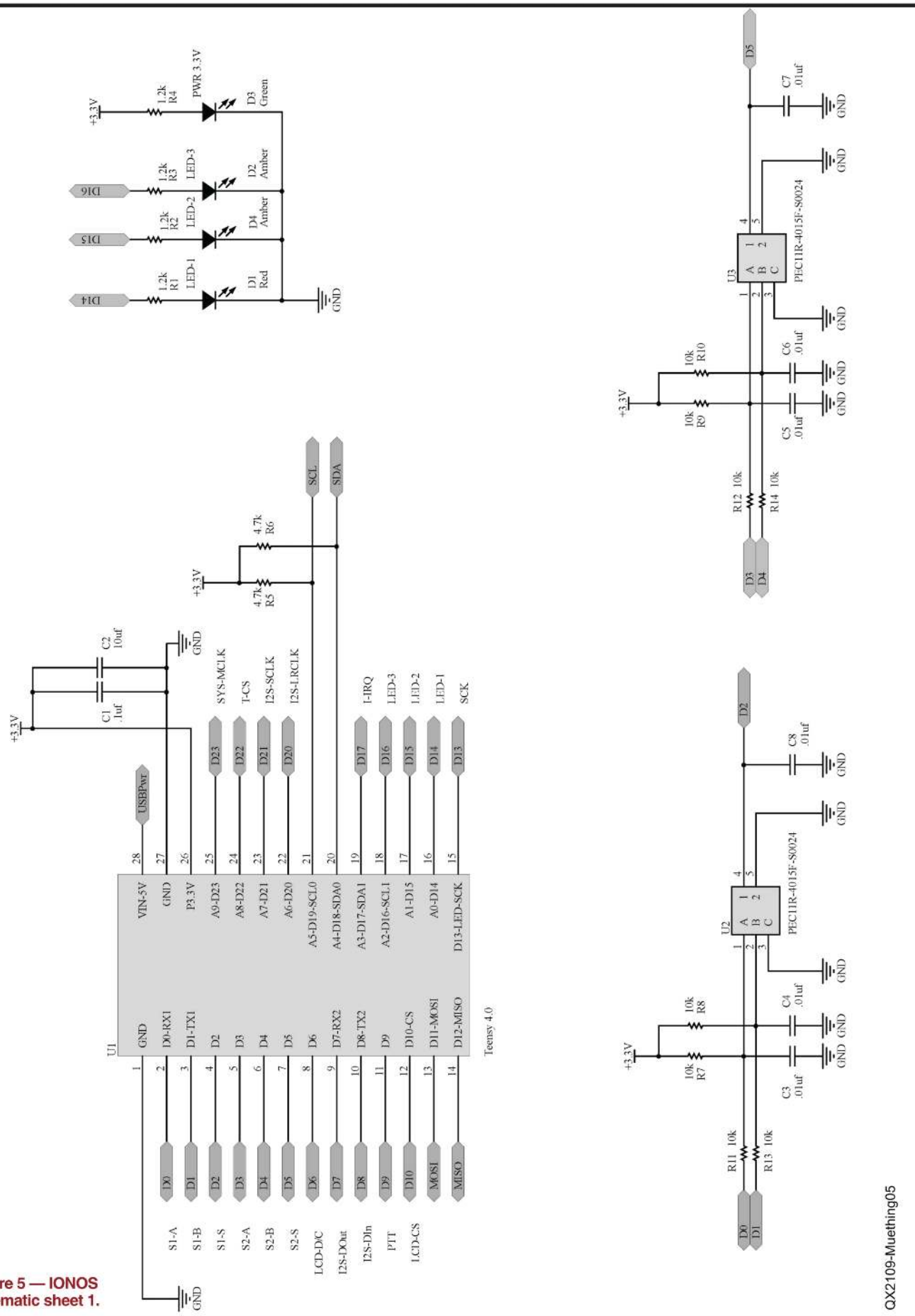


Figure 5 — IONOS schematic sheet 1.

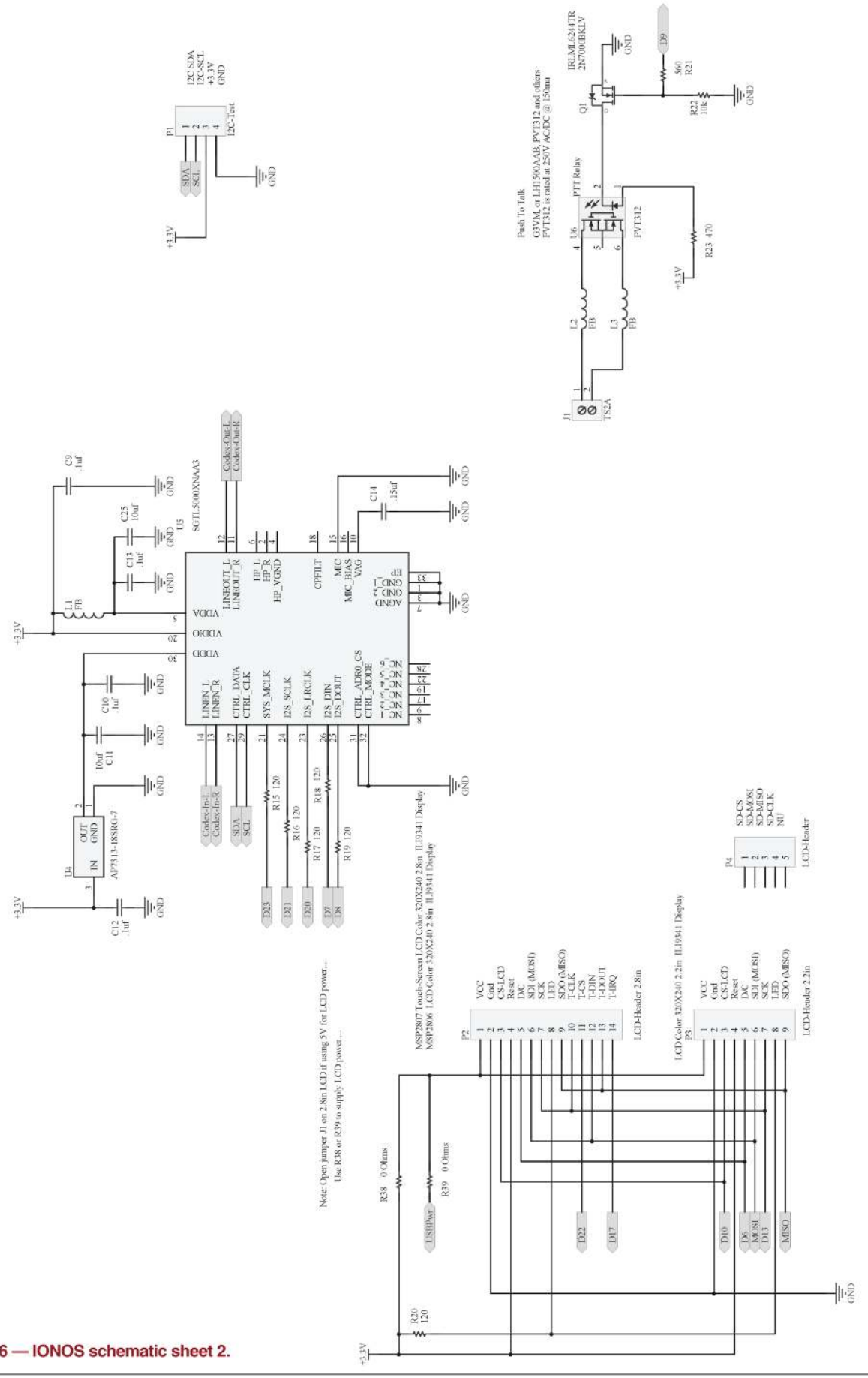


Figure 6 — IONOS schematic sheet 2.

QX2109-Muething06

driver chip with readily available driver support in the Arduino environment. The back-light LED drive can be supplied via 3.3 V or 5 V from the Teensy USB power pin by selecting one of two 0 Ω resistors on the board. Also note that you must cut a jumper on the LCD display if you are using 5 V, see the LCD display data sheet.

The input op-amps are connected as a buffer in the basic configuration, with a 0 Ω resistor connecting the minus input to the output. It can change to provide a +12 dB gain by changing the 0 Ω resistor to 4.2 kΩ and adding the 10 kΩ and 10 µF capacitor to the minus input. The software and parameter encoders provide an input gain adjustment range of 1 to 200 in 1, 2, 5 steps.

The input and outputs are via standard stereo 1/8 in (3.5 mm) audio jacks. The input is terminated into to a 100 kΩ resistor, with a 10 µF capacitor to provide dc isolation, to the + input of the op-amp. Two 100 kΩ resistor provide a half of the 3.3 V as bias on the op-amp input to give symmetrical operation. The outputs from the codec are via a 10 µF capacitor into a 120 Ω resistor to a 1/8 in (3.5 mm) stereo audio jack.

All design schematics, Altium Gerber files, and mechanical files are located on our GitHub page as are details on how to setup the IDE development environment. **Figures 5, 6,** and **7** show the IONOS Simulator schematic sheet. **Figure 8** is a picture of the completed packaged simulator showing screen and USB and audio jack placement.

Simulator Software

The simulator and user interface are completely controlled by the Teensy 4.0 microprocessor and most audio processing is handled by the Teensy Audio Library acting on the 16 bit 44.1 kHz sample rate stereo audio. The software was developed on the simple and free Arduino platform (available for Windows, MacOS and Linux). Approximately 2,200 lines of C++ code implement the Simulator, user interface, serial interface and interfaces to the encoders and TFT display. All the normal audio processing including high- and low-pass FIR filtering is done using the Teensy Audio components and library. The Very Low Frequency (VLF), <3 Hz, low-pass filters, which must meet CCIR/ITU specific bandwidth and response curves, are implemented in *separate* 32-bit-precision floating-point FIR implementations that process at a sample rate of 64 times the selected channel Doppler spread (2 sigma values shown in **Figure 1**). Originally IIR filters were used for these VLF filters

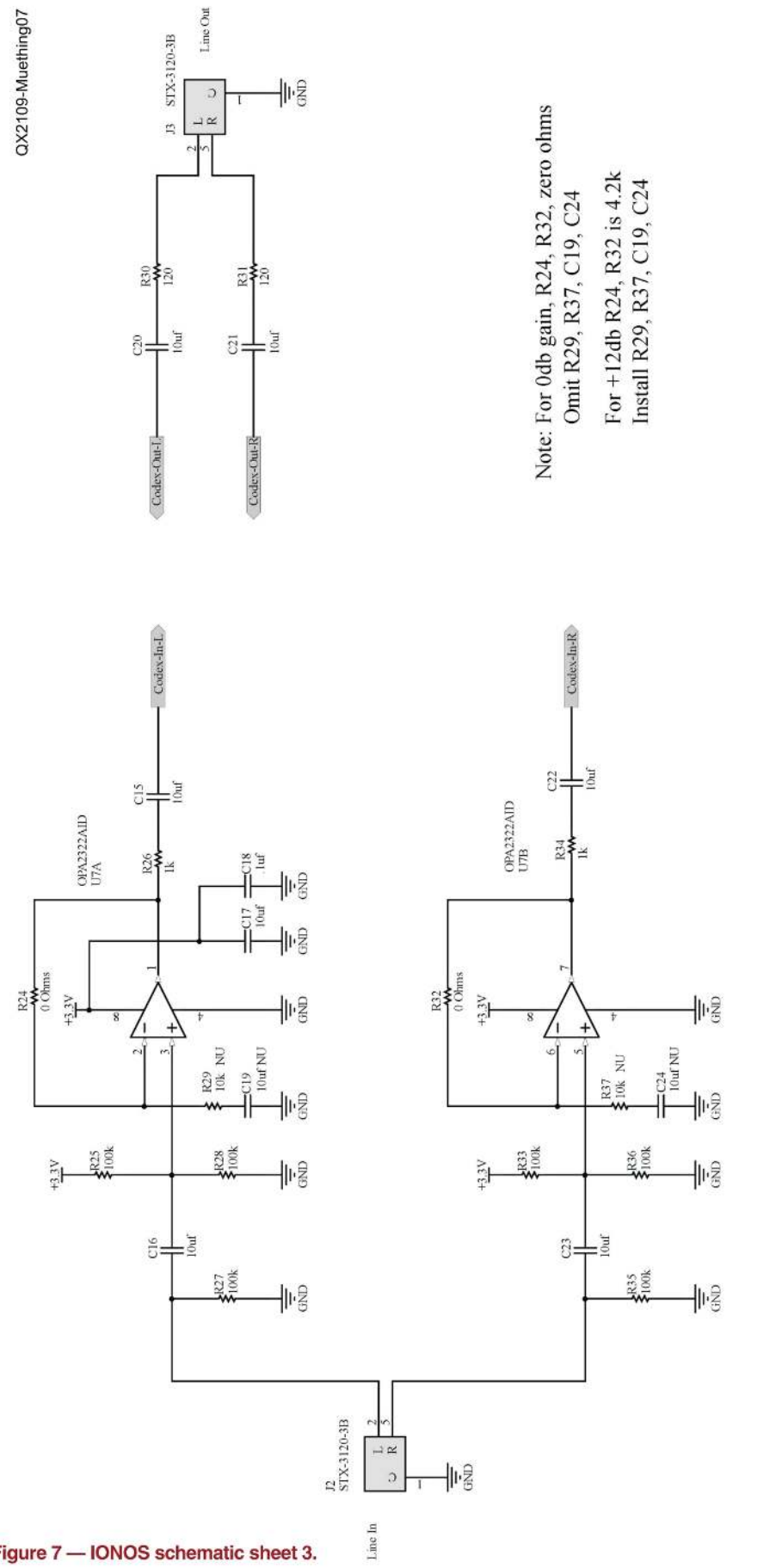


Figure 7 — IONOS schematic sheet 3.

but during verification, stability concerns and precision requirements dictated better performance and stability was needed and these filters were changed to modest 128 tap FIR filters, see **Figure 9**.

Using the Simulator

The simulator is used in the audio path(s) connecting one "station" to another. No radios are used or required. The Simulator creates audio distortion in a precise and "statistically-standardized" way that simulates what would happen due to the RF distortion when the radio wave is passed through the ionosphere on one or more paths at the selected signal to noise ratio (SNR) between –40 and +40 dB. **Figure 10** shows typical connections for a half-duplex modeling using one simulator. The modems could be hardware, firmware, or software DSP (sound card) implementations and are the same as used to connect to the radio's audio inputs and outputs.

Example Simulations

As part of the testing, debugging, and verification of the simulator, we encouraged some interested beta testers to run tests of real-world amateur protocols and modems using the standardized CCIR/ITU channels. This was compared when possible by runs on other software or hardware simulators to compare and confirm results. Runs were typically between 5 and 15 minutes per run at various SNR values for each channel type. Because of the randomization used in the path model and the complexity of adaptive protocols there is always some variation of measured net throughput between runs.

Simulator Results for Various Winlink Digital Modes

As part of testing the simulator, we decided to systematically evaluate the various Winlink digital modes figuring the results would be useful to users deciding which modes were best under what circumstances. For HF ARQ protocols PACTOR 2, 3 and 4

Photo of First prototype production Run

Size: 6.1" x 4.7" x 1.4" (155 mm x 120 mm x 36 mm) (excluding knobs)

Figure 8 — First prototype production run with enclosure.

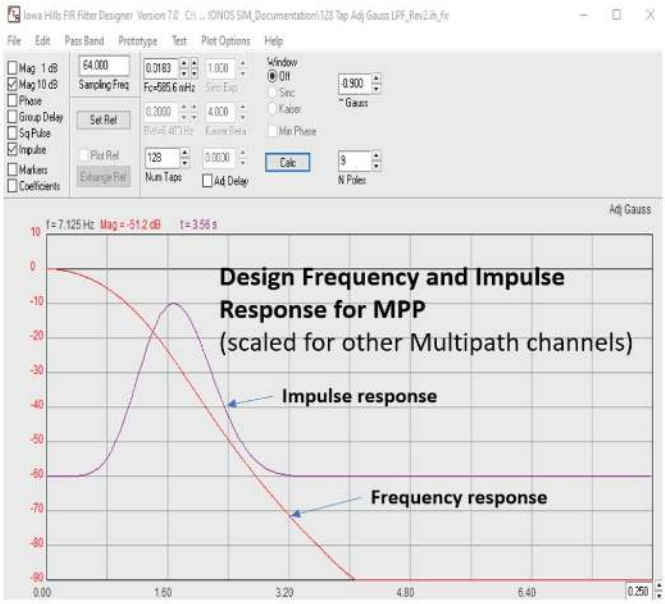
~Gaussian 9 Pole LP FIR Filter:

(used to narrow band filter Gaussian Noise for multipath processing)

Sample rate: 64 x Doppler Spread (2x CCIR recommended minimum)

Follows CCIR guidelines for Bandwidth and roll off.

Iowa Hills FIR Filter Designer Version 7.0



High resolution Spectrum Analyzer measured Response for MPP
(scaled for other Multipaths)

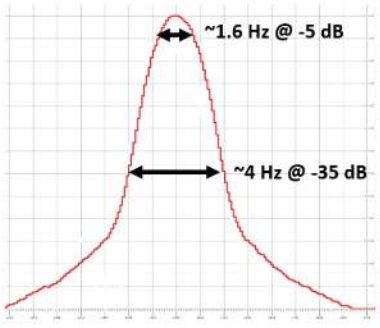


Figure 9 — Gaussian 9 pole LP FIR Filter.

were evaluated as were sound card modes WINMOR, ARDOP and VARA HF 2300. We also simulated VARA FM and AX.25/FX.25 packet for a VHF channel.

The first example, **Figure 11**, shows results for wideband modes PACTOR 3 and 4, VARA HF, ARDOP 2000 and WINMOR 1600 with 3 kHz Additive White Gaussian Noise (AWGN).

PACTOR 4 clearly beat the other modes over the realistic SNR range likely to be encountered. VARA 2300 did very well edging out PACTOR 3 for SNR better than 17 dB. The older ARDOP and WINMOR modes trailed badly versus the other modes. PACTOR 4 (1.9 kHz) uses less bandwidth than VARA (2.3 kHz) or PACTOR 3 (2.4 kHz). PACTOR 4's higher symbol rate with automatic path equalization delivers over 70% throughput improvement in typical multipath channels. PACTOR 4 cannot currently be used in the United States due to an FCC symbol rate restriction. Imagine what new sound card modes operating without this restriction might achieve.

Figure 12 shows how these same modes stacked up with multipath "good" (MPG) conditions simulated. Results are similar with PACTOR 4 having dominance across the entire range of conditions. As a final HF example, **Figure 13** shows how performance looked for multipath "poor" (MPP) conditions. Again, the same relationship between the modes persisted in this pessimistic HF channel.

Finally, **Figure 14** shows the results for VARA FM versus AX.25 and FX.25 packet. VARA FM is an exciting new digital mode with much better performance than traditional packet. It is so much faster that we had to use a log scale to depict the difference. FX.25 is a superset of the AX.25 with additional overhead for including forward error correction information. FX.25 did work at SNR levels that AX.25 would not achieve but VARA FM was never slower than FX.25 even under very poor conditions.

The complete results can be found in Tom Whiteside's (N5TW) paper "A comparison of Winlink digital mode performance based on simulation results using the Teensy IONOS Simulator – Take Three **[8]**.

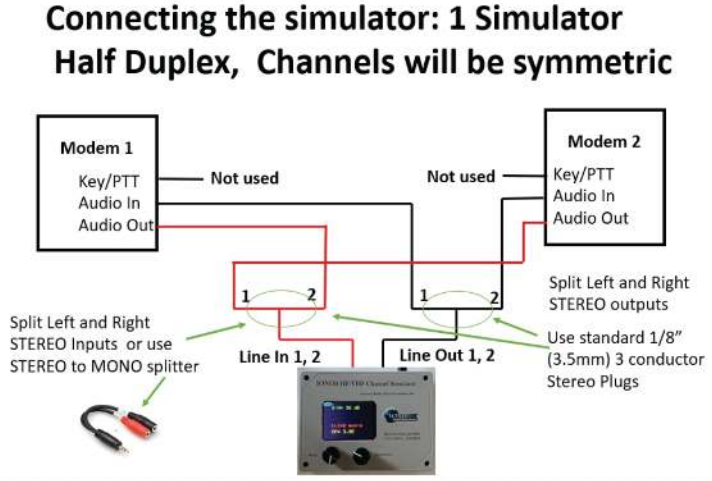
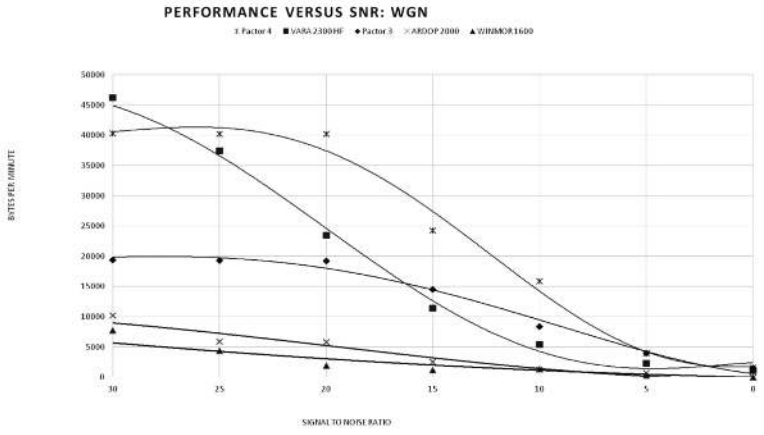


Figure 10 — Modem connections to the IONOS Simulator for half duplex ARQ modeling.



QX2109-Muething11

Figure 11 — Performance versus SNR for 5 ARQ protocols in Additive White Gaussian Noise (AWGN).

What we Learned

There were of course minor PCB and mechanical issues, software bugs and also some confusion of how to correctly implement and calibrate the Watterson model. Sections of the code that processed the VLF IIR filters for the path modeling had to be redone as FIR filters with higher precision. We also had to verify calibration against other simulators and compare results using the standard CCIR channels at the same SNR levels. Not expected but beneficial results were modifications to the modem client program (Winlink Express) that were needed to keep the Pactor4 modem buffered to achieve full speed and some firmware modifications to the Pactor4 modem. The sum of these changes contributed to approximately a net 15% improvement in Pactor 4 net throughput for several simulation scenarios. Those kinds of detailed and repeated simulations would have been impossible with conventional over-the-air or basic WGN simulations.

Contributors

We would like to thank all those participating in the project, especially Hans-Peter Helfert, DL6MAA, and Phil Sherrod, W4PHS. Other beta testers included Phil Karn, KA9Q, David Rowe, VK5DGR, and Gordon Gibby, KX4Z.

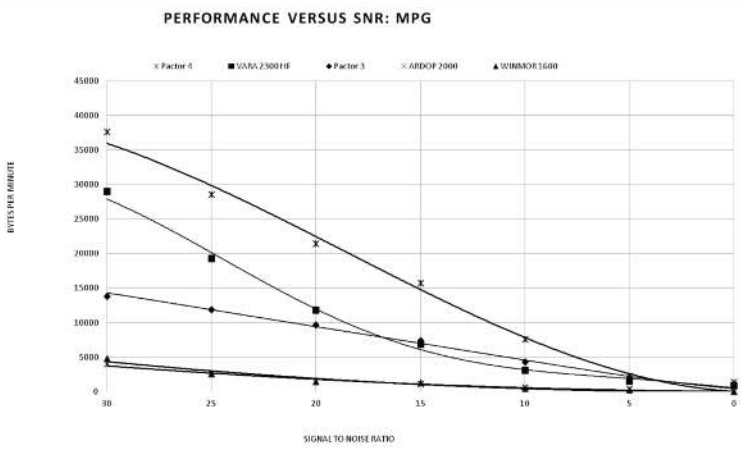
Availability

The GitHub site **[8]** contains all the information and documentation necessary to duplicate the simulator. The ARSFI will provide a limited number of completed and tested simulators and surface mount PC boards to radio clubs or individuals on a first come first served basis. Ordering information for complete units and boards is available at **https://winlink.org/content/ionos_simulator**.

Rick Muething, KN6KB holds an Advanced Class license, and was first licensed in 1962. He graduated the University of Cincinnati BSEE Honors, 1969 and Northeastern University MSEE High Honors 1972. Raytheon and Honeywell Aerospace computer design. Senior engineer (Military Computer Design) at Applied Technology/Itek Sunnyvale, CA. Co-founded Paratronics Inc. (Logic Analyser Instruments) as VP Engineering in 1976. Co-founded startup Asix (ASIC chip tester) and went public as Credence in 1992. Part time Graduate school faculty at Northwestern Polytechnic Univ. Freemont, CA. 1992-1995. Consultant to Teradyne (IC Chip Testers) 1996- 2002. Board member and Sect/Treas. of Amateur Radio Safety Foundation. 1997- present Member of Winlink development team doing software and protocol development.

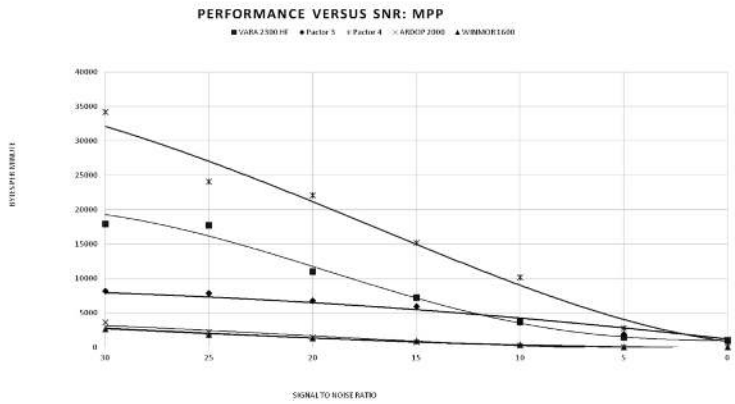
Tom Lafleur, KA6IQA, holds a General Class license, and was first licensed in 1975. Taught computer technology at the college level. Co-Founder GNAT computers an early maker of Intel 8008, 8080 and Z80 systems. Member of technical staff US Naval Electronics Laboratory (NEL) in the Radio Propagation Group's, "La Posta Astro-Geophysical Observatory" 18M radio Telescope facility. VP Engineering and information technology M/A-COM-Linkabit. Director of R&D and VP information technology Digital Research (CP/M). VP Engineering and information technology QUALCOMM and a member of its engineering management team. Co-Founder and CTO Rhythms Net-Connection, an early DSL company. Co-founder, CTO and board member of DriveCam now Lytx, a driving safety company. I am now mostly retired but continue as a board members for a number of companies and continue doing system engineering consulting. I am a member of the Winlink Development Team and on its board of directors.

Tom Whiteside, N5TW, was first licensed as a Technician-plus in November 1995 and upgraded to Amateur Extra Class in April 1996. He received a Bachelor of Science in Electrical Engineering from the University of Texas at Austin in May 1975. He worked as an engineer for the Motorola Government Electronics Division and the Motorola Microprocessor groups from 1975 to 1982. He worked as an IBM development engineer / manager from 1982 to 1993 in microprocessor and systems architecture



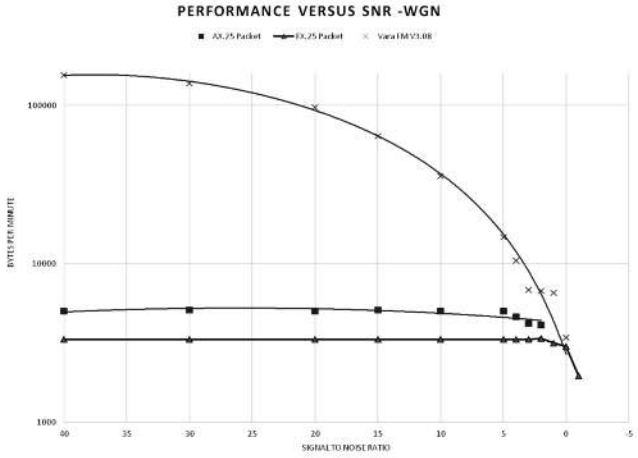
QX2109-Muething12

Figure 12 — Performance versus SNR for 5 ARQ protocols in Multipath Good (MPG).



QX2109-Muething13

Figure 13 — Performance versus SNR for 5 ARQ protocols in Multipath Poor (MPP).



QX2109-Muething14

Figure 14 — Performance for VARA FM and AX.25_FX.25 Packet.

RF Connectors and Adapters

DIN – BNC
C – FME
Low Pim
MC – MCX
MUHF
N – QMA
SMA – SMB
TNC
UHF & More

Attenuators
Loads & Terminations
Component Parts
Hardware
Mic & Headset Jacks
Mounts
Feet – Knobs
Speakers & Surge Protectors
Test Gear Parts
Gadgets – Tools

www.W5SWL.com

and was the first co-director of the Somerset PowerPC alliance with Motorola and Apple. In 1993, he moved to California as President of MIPS Technology, Inc where he was a Silicon Graphics Vice President. In 1995, he returned to IBM as a Vice President responsible for Austin Microprocessor development including the PowerPC alliance. He retired from IBM in 1995 as part of a long-term goal to retire on his 45th birthday.

References

[1] C. C. Watterson, J.R. Juroshek, and W.D. Bensema, "1970 Experimental confirmation of an HF channel model," IEEE Transaction of Communication. Technology. Vol COM-18. pp 792-803 Dec 1970.
[2] CCIR/ITU Rec. 520-2 1 9E2c: "Influence of the ionosphere Recommendation 520-2 Use Of High Frequency Ionospheric Channel Simulators," (1978-1982-1992).
[3] J. B. Forrer, KC7WWA, "Low-Cost HF Channel Simulator for Testing and Evaluating HF Digital Systems," ARRL/TAPR DCC 1999.
[4] **https://www.pjrc.com/teensy/**.
[5] **https://www.pjrc.com/teensy/td_libs_Audio.html**.
[6] TFilter: **t-filter.engineerjs.com/**.
[7] Iowa Hills Software: **www.iowahills.com/**.
[8] **https://github.com/ARSFI/HFSimulator**

Upcoming Conferences

New Mexico TechFest 2022

February 26, 2022

Virtual Event

www.rmham.org/new-mexico-techfest/

Rocky Mountain Ham Radio, New Mexico is pleased to announce the 2022 New Mexico TechFest is Saturday February 26, 2022.

Join fellow amateur radio operators from around the Land of Enchantment (and beyond) for a day of quality presentations, demonstrations, and instruction provided by some of New Mexico and Colorado's most experienced technical hams on a variety of emerging and relevant technical topics within amateur radio today. The New Mexico TechFest is designed to provide a unique opportunity for all hams interested in the technical aspects of our hobby to advance and expand their technical knowledge and to facilitate technical discussion, collaboration, and discuss ideas with one another.

SCALE 19x

March 3 – 6, 2022

Pasadena, California

www.socallinuxexpo.org/scale/19x

SCALE 19x, the 19th annual Southern California Linux Expo, will take place March 3 – 6, 2022, at the Pasadena Convention Center. SCALE is the largest community-run open-source and free software conference in North America. It is held annually in the greater Los Angeles area. See website for details.

5th Annual Amateur Radio Utah Digital Communications Conference

March 26, 2022

Sandy, Utah

www.utah-dcc.org

The Utah Digital Communications Conference will be held Saturday March 26, 2022. We are looking for presentations on all things digital in ham radio, specifically presentations around using raspberry Pi or Arduino type computers with ham radio. Hot Spots, DMR, D-STAR, HF digital modes, working satellites, Mesh, APRS, Winlink, building your own battery packs, etc. We will also have some display tables where you can display projects, etc.

Society of Amateur Radio Astronomers Spring Conference

April 16, 2022

Virtual Event

www.radio-astronomy.org

The 2022 SARA Spring Conference will be held on Zoom, April 16, 2022. This virtual conference will replace the annual SARA Western Conference because of the continuing COVID-19 pandemic.

Papers and presentations on radio astronomy hardware, software, education, research strategies, philosophy, and observing efforts and methods are welcome. See website for paper submission details and registration information.

Arlen Young, K6KZM; 250 Scripps Ct., Palo Alto CA 94306-4540; **Arlen_young@yahoo.com**

Compact Directional Low-Band Receiving Antenna

An array of two phase and amplitude matched loop antennas works well for the 160 to 40 meter bands.

Figure 1 — The two loop antennas in front of the author's home. The loop to the right has a strap for adjusting signal amplitude.

My house is situated on a small suburban lot at the base of the San Francisco Peninsula, so my possible antenna sizes are limited. Several years ago, I became interested in the 160 meter band. After achieving my Worked All States award, I set my sights on the 160 m DXCC award. Because I am surrounded by high density cities, the noise on 160 m is considerable. For a long time, I used my vertical antenna for receiving. Then, I calculated that a loop antenna would reduce noise uniformly distributed across all azimuthal angles by 3 dB. In addition, it could be used to null out localized interference sources, so I purchased a Wellbrook loop antenna [1].

I found the loop to be quieter than the vertical antenna. I could copy signals that were buried in the noise on the vertical antenna. Not surprisingly, I found by rotating the loop and minimizing the noise, that most of the noise was coming from the directions with the greatest population.

I was inspired by an article describing the phasing of two loop antennas to further reduce noise [2]; the loop separation was only 20 ft. I did some trigonometric calculations, and saw that the noise reduction could be significant, so I purchased a second loop. In this article, I describe how I combined the two loops, and the performance results. An advantage of the design is that it works well for several of the low bands: 160, 80, and 40 meters in particular, and without having to change anything when the bands are changed. The signal reception direction is also easily reversed in the shack.

While I chose the Wellbrook loop antennas, the description for setting up the phased loop system presented here can also be applied to other phased arrays.

Design

Figure 1 shows the two loops on the edge of the sidewalk in front of my house. I found that when the loops were placed several feet from the house, the noise level was about 3 dB higher. The two loops are 3 ft above the ground and oriented in the plane containing the desired direction of signal reception. Fortunately, my sidewalk is in line with the path to Europe, my primary DX interest. The two loops are also oriented 180° from each other. The BNC sockets on the Wellbrook preamplifier modules face either toward or away from each other. Each loop has a feeder into the shack, and the feeder

lengths are the same. The two signals at the shack-ends of the feeders are nominally out of phase with each other.

Figure 2 shows a block diagram of the complete antenna system. Inside the shack, the two feeders are connected to the Wellbrook antenna interfaces (**Figure 3**). These modules supply power along the feeders to the loop preamplifiers. The outputs of these modules are connected to the inputs of the signal combiner **[3]**. The vector addition of the two signals is performed by the combiner. The output of the combiner is connected to the receiver input. It is important that transmitter power is never applied to the combiner.

A delay line is inserted into the signal path of Loop 2. For convenience, I inserted the delay line between the antenna interface module and the combiner input. The electrical length of the delay line is equal to the separation of the two loops. Consequently, signals arriving from the reverse direction will be seen at the combiner as exactly 180° out of phase and cancel out in the combiner. Signals arriving in the forward direction will first be received by Loop 1, and then after the 20 ft propagation delay, by Loop 2. The delay line creates an additional 20 ft of delay, so that the two signals arriving at the combiner are delayed from each other by the equivalent of a propagation distance of 40 ft. The vector sum of the two signals is non-zero.

Loop Orientation

Normally, the two loops are oriented in the plane containing the desired signal direction. However, in some cases of a strong, localized interference source, the two loops can be rotated to null the interfering signal. For example, from the azimuthal single-loop pattern (data sheet in **[1]**), it can be seen that rotating each loop up to +25° from the line between to the two loops will not reduce the signal amplitude by more than 1 dB. Both loops must be rotated to the same orientation.

Feeders

The two feeder lines must be the same electrical length. I used LMR-240UF coax. I cut a length of coax to reach from the shack to the more distance loop. Leaving one end open, I measured its resonant frequency. Then I cut a second length of coax with the same physical length plus several inches for pruning. I pruned the second coax until it had

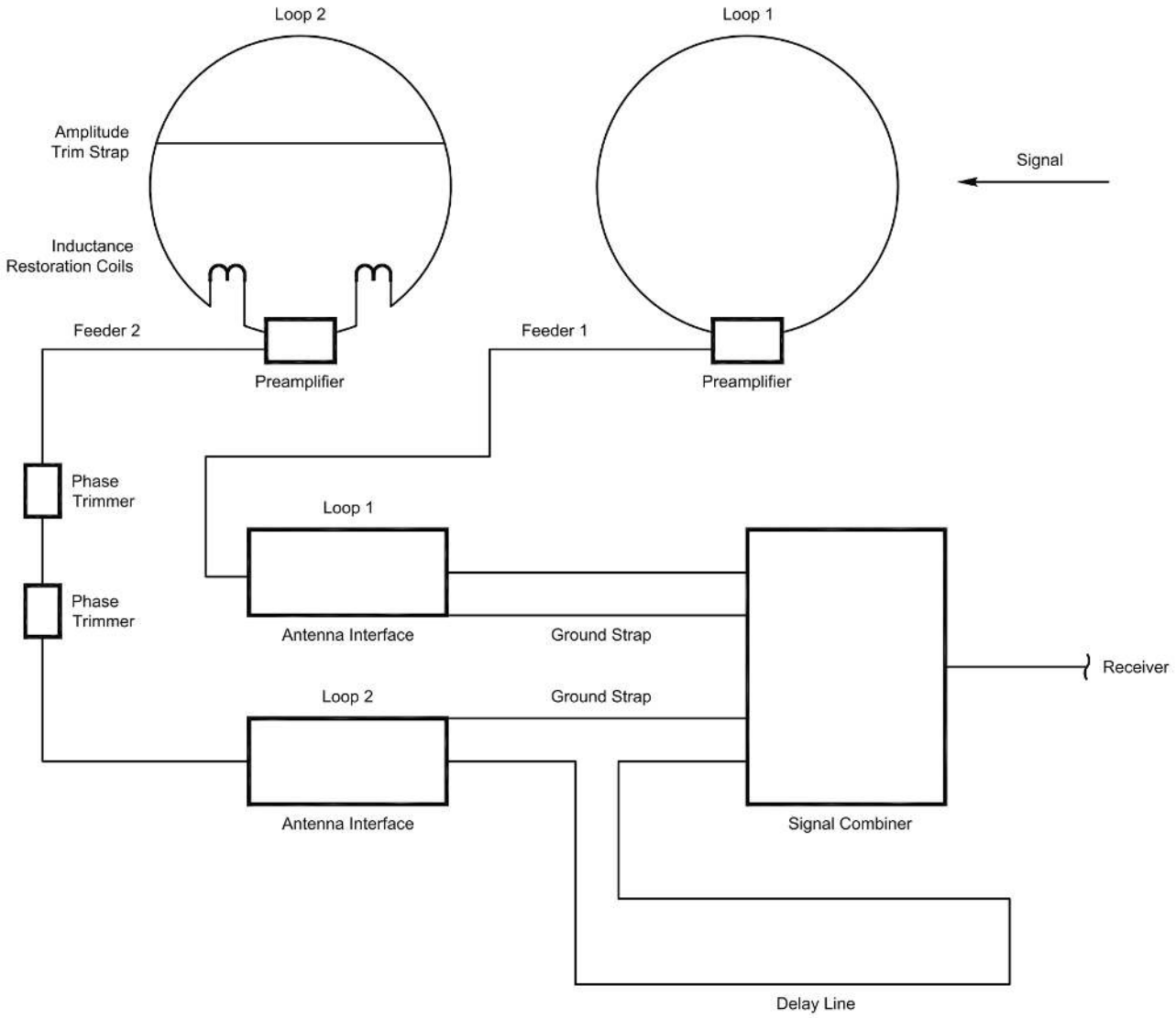


Figure 2 — Block diagram of the phased loop system.

the same resonant frequency as the first coax.

Amplitude Matching

For the deepest null of signals from the reverse direction, the signal amplitudes from the two loop preamplifiers should be identical. I used an AM radio station at 1640 kHz and 46 miles away to measure the signal amplitudes. It provided a very steady S-9 +11 dB signal. I found that the signals from the two loops differed by 2 to 3 dB. I reasoned that the easiest way to reduce the stronger signal was to reduce the effective capture area of its loop. I did this by connecting an adjustable strap across the loop. The strap is two telescoping aluminum tubes, secured with a hose clamped. A 1-inch long strip of sheet aluminum is screwed onto each end of the strap. The two strips are secured to the loop with hose clamps. After loosening the hose clamps, the strap can be slid up and down the loop, and its length will decrease and increase. The strap can be seen on the loop at the right of **Figure 1**, 17 inches below the top of the loop.

When observing the two signal amplitudes, I inserted 18 dB of attenuation in the receiver to move the S-meter needle to the lower one-third of the scale. The S-meter scale expands toward the low end, making small differences in amplitude easier to see.

I moved the strap down the loop until both loop signal amplitudes were the same. In doing so, the inductance of the loop decreased, causing a phase shift in the signal. To restore the inductance back to the value for the original loop, I inserted two coils between the ends of the loop and the inputs to the preamplifier. These coils are about three turns, 1 inch in diameter, and mounted with vertical axes to minimize signal pickup, see **Figure 4**. They are wound in the same sense, so that any signal they pick up will cancel out at the differential input of the preamplifier.

Each coil has one-half the inductance lost by attaching the strap. To determine the coil inductance, I removed the strap and the preamplifier leads from the loop and measured its inductance. I then reattached the strap and measured the inductance again. Each coil needed to have an inductance of one-half of the difference between the two measured values. I wound the two coils and attached them to the loop and preamplifier leads.

Restoring the loop inductance unfortunately also changed the signal amplitude. The strap and coils are interrelated. I began the tedious process of changing the coils by 1/8 turn for each iteration and repositioning the strap until the signal amplitudes matched again. After several iterations, I was able to obtain an exact amplitude match and the original loop inductance, as seen by the preamplifier.

My primary interest is working European countries on 160 meters, so my array is oriented in that direction. However, the Wellbrook loops are light weight, so mounting them on a rotating boom is feasible.

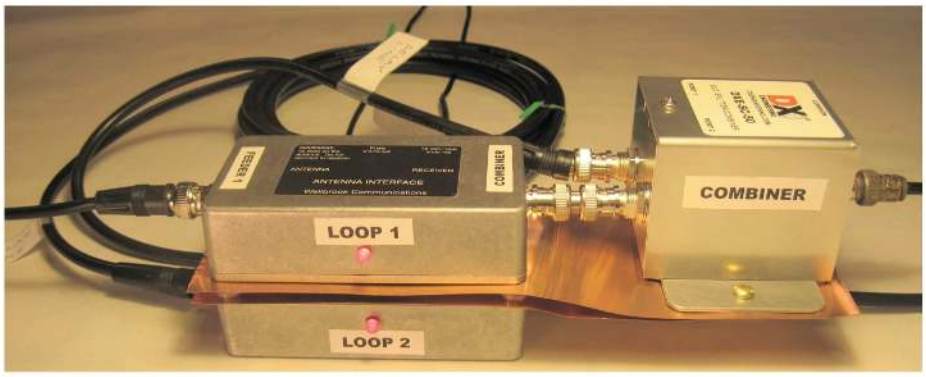


Figure 3 — Antenna interfaces, signal combiner, and delay line.

Figure 4 — Inductors inserted between the loop and preamplifier to compensate for the loop inductance reduction caused by the signal amplitude-reducing strap.

Phase Matching

To match the phase of the two loops, preamplifiers, and feeders, I again used the AM radio station as a signal source. I placed the two loops about 20 ft apart to minimize interaction. I oriented them facing each other, so their signals would cancel, and with collinear axes. Also, the loops were placed such that their axes were perpendicular to the path to the radio station. The path can be determined by nulling the radio signal with one of the loops. With both loops connected, I watched the signal null at the receiver as I inserted various short lengths of coax, first into one signal path and then the other. I found that indeed there was a small phase difference between the two paths. About 3 ft of coax inserted into one of the paths produced the deepest null of about −33 dB.

Delay Line

The delay line is a section of coax having an electrical length equal to the electrical distance between the two loops. I measured the velocity of propagation of my LMR-240 coax to be 0.784, so the physical length of the coax is about 16 ft. I cut the coax a little longer to enable pruning to the correct length. The resonant frequency of the open-ended coax corresponding to an electrical length of 20 ft is 12.301 MHz. I pruned the coax to this resonant frequency and inserted it into the signal path of Loop 2, as shown in **Figure 2**. Reception direction can be reversed in the shack by switching the delay line into the signal path of Loop 1.

The inductance of the shield of the coiled delay line initially created a ground shift

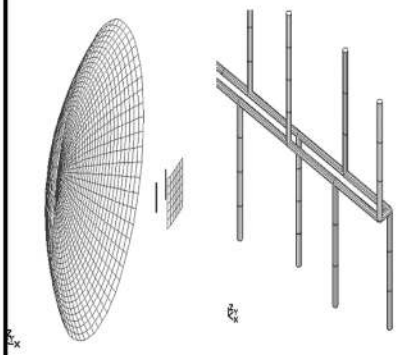
between the two antenna interface modules. This ground shift caused a degradation of the front-to-back ratio. A copper strap between each module and the combiner eliminated this shift. Each strap is 3 inches wide copper foil **[4]**, and is attached to the module by the four screws securing the bottom plate. As seen in **Figure 3**, the module for Loop 2 was inverted to facilitate connecting the strap to the combiner. Consequently, the module BNC sockets are reversed from the **Figure 2** drawing.

ANSim

Antenna Modeling Software

Affordable Precision

ANSim is an extremely accurate moment method program for modeling antennas and other radiating structures.



- **All Versions**
- Use of multi-radius wires
- easy modeling of coax
- Double precision accuracy
- Segments up to ¼ wavelength
- Export information in multiple formats
- Compatible with NEC BSC
- **Plus ad Pro Versions**
- 5,000 &40,000 segments respectively
- 2D surface patch elements,
- Use of dielectrics

EM-Bench

www.em-bench.com

Phoenix Antenna Systems Ltd.
www.phoenixantennas.com

Noise

Minimum loop spacing is determined by the noise floors of the receiver and loop amplifiers. The differential signal from the two loops decreases as the spacing is decreased.

I checked the receiver noise floor by connecting a 50 Ω termination to the receive antenna input (RX). This termination was supplied with the combiner. With the Yaesu FTdx5000MP receiver gain set to maximum (AMP2 ON) and a bandwidth of 3 kHz, no noise was detectable on the S-meter.

The noise floor of the loop preamplifiers was determined by disconnecting one of the loops from its preamplifier and replacing it with a toroidal inductor having the same inductance as the loop. A toroidal winding was used to reject pickup of any ambient signals and noise. The combined receiver and preamplifier noise was barely detectable on the S-meter. The typical 160 meter noise at night received by the loop array is many times greater. Therefore, the 20 ft loop separation suggested by Roberts **[2]** seems to be adequate. For systems with a higher noise floor, the loop spacing and delay line length can be increased to improve the signal-to-noise floor ratio.

Although the Wellbrook loop is not shielded, the preamplifier responds only to the currents induced in the loop by signal magnetic fields. Electric fields, which can induce significant noise into vertical antennas, are rejected in common mode by the balanced differential inputs of the loop preamplifier.

Noise Reduction versus a Single Loop

For noise sources uniformly distributed in azimuth, the dual loop array rejects 4 dB more noise than a single loop. For the elevation plane, I assumed that signals with arrival angles between 0 and 60° horizontal elevation are acceptable, and signals with arrival angles of 60° to 180° are to be rejected. For noise sources uniformly distributed across the rejection sector, the dual loop array rejects 8 dB more noise than a single loop. I derived these rejection numbers from trigonometric calculations.

I measured the front-to-back ratio using the AM station cited above, and by exchanging the delay line across the two signal paths. I observed a 26 dB front-to-back ratio.

Summary

Since installing this loop array, I have worked many new countries on the low bands. I am very pleased with its performance. The Wellbrook loop antennas and my Yaesu FTdx5000MP work well together, but of course other loops and receivers can be used as well. When constructing this loop antenna array, the following should be considered for best performance.

• *The signal amplitudes from the two loops and preamplifiers must be matched.* Even preamplifiers of the same model can have significant gain variations, so the amplitude differential across the two signal paths should be checked and minimized. I found it easiest to reduce signal amplitude by reducing the capture area of the loop. But other means can also be used, such as attenuators, as long as reflections are not created in the signal path.

• *The phases of the two signal paths must be matched.* Even when the coax cables are cut to the correct electrical lengths, there can still be a small but significant phase differential.

• *The noise floor for the combined receiver and preamplifiers should be less than the received background noise.* The relative noises are worth checking because, due to its compact size, the signal amplitude from the loop array is much lower than from a single loop or vertical antenna.

[Photos by the author].

Arlen Young, K6KZM, is retired with a BS from Stanford University and a PhD in experimental quantum physics from the University of Illinois. Prior to retirement, Arlen worked for 37 years in Silicon Valley industry as an electronics engineer and scientist. He was one of the pioneers in developing UPC bar code scanning, and holds more than 50 patents relating to computer data storage. Arlen, first licensed as KN6KZM in 1955, now holds an Amateur Extra Class license and is an ARRL Life Member. He enjoys working DX on 160 m through 6 m using FT8, CW, and SSB. He holds the 5 Band DXCC award and the DXCC Challenge award.

References

[1] Wellbrook ALA-1530LN loop antenna with antenna interface module.
[2] M. Roberts, KK5JY, "Small Loop Antenna Array for HF Reception," 12/15/2018; **www.kk5jy.net/rx-loop-array/**.]
[3] DXEngineering DXE-SC-50 50 Ω signal combiner.
[4] Copper foil available from garden supply or **www.amazon.com**.

Eric P. Nichols, KL7AJ

AlasKit Educational and Scientific Resources, 3763 Lyle Ave., North Pole, AK 99705; **kl7aj72@gmail.com**

Self-Paced Essays — #10 Almost AC

Essay 10: Investigating ac electronic behavior.

Now that we have established a reasonably firm foundation for dc electronics, we can smoothly approach ac electronics. We trust Ohm's Law, Kirchhoff's Laws, and the rudiments of network analysis are becoming friendly and familiar tools. Before we dive right into ac electronics, however, we will explore what I call *"Almost AC"* electronics, that is, the behavior of electrical circuits under a changing state. We will explore the all-important RC time constant, and the nearly-as-important L/R time constant.

RC circuit

Let's toss together a very simple series circuit (**Figure 1**) consisting of a voltage source, a switch, a resistor, and a capacitor.

We don't need to assign any component values yet. Let's just look at some "self-evident truths" about this circuit. Harking back to our earlier essays, we can learn a lot with simple plumbing analogies; they aren't perfect, but they're certainly *useful.* Let's imagine capacitor C1 as a storage tank for electrons. Let's add a little detail to our storage tank. We can assume it is a uniform diameter cylinder, standing on end, and fed *from the bottom* with a hose. The hose itself has some finite diameter; it has a fixed opposition to water flow, the mechanical equivalent of electrical resistance.

Our task is to fill up the tank from the bottom through our "resistive" hose. When the tank is empty, the tank itself presents no opposition to the flow of water. As the tank begins to fill, however, there is a certain amount of back pressure generated, by the weight of the water attempting to drain back down into the hose. As the tank continues to fill, the increased weight of the water increases the back pressure. If the cylinder is of uniform cross section, and we're using a non-compressible fluid — like water — the weight of the water will increase in a linear manner. One nice thing about electronics is that electricity is a totally *non-compressible* substance, which *greatly* simplifies matters!

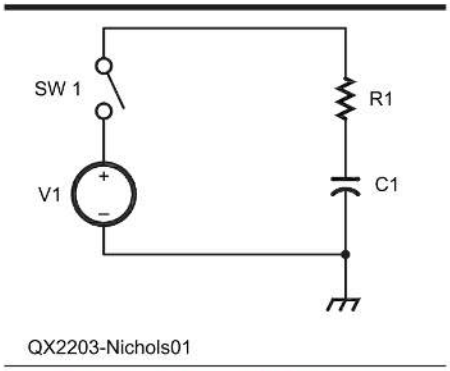


Figure 1 — A simple RC network.

Because the hose connecting the pump to the tank has some resistance, we find that the tank does not continue to fill at a uniform rate. In fact, we find that the rate of water flow into the tank decreases in proportion to how full the tank is.

We can now logically (almost) replace the water pump with our voltage source, our skinny hose with the resistor, and the water tank with the capacitor. Now, we do need to make a couple of concessions in "translating" plumbing into electrical circuits. First of all, the "resistance" of a water hose is not constant; in fact it increases with the increase of water flow. The resistance of an electrical resistor is assumed to be constant — if we disregard heating effects. A water hose is actually more closely equivalent to a constant current source, which we will talk about a bit later. So, we need to make this adjustment in our thinking. Second, an electrical circuit has to be, well, a *circuit.* We can't just have electrons flowing out the end of a wire, and furthermore a "one ended" capacitor doesn't exist. But if we disregard — or at least, forgive — these discrepancies, we can make some progress.

Returning to our electrical circuit, we can substitute the *pressure* of the water at the bottom of the tank with the *voltage* at the "top" of the capacitor. When fully discharged, our capacitor has the least opposition to current flow...in fact, any fully discharged capacitor looks like a dead short at the instant the voltage is applied. As the capacitor "fills up," its opposition to current flow increases, while the voltage across said capacitor continually increases. It will continue to increase until it reaches the same voltage as the voltage source, at which time current ceases to flow. Well, almost. Actually, the capacitor voltage never reaches the supply voltage, but approaches it *asymptotically.*

Before we forget, let's look at the resistor itself. When we first close S1, the current through R1 is at a maximum, the current being limited *only* by R1. Remember, at this instant, C1 looks like a dead short. Of course, being a series circuit, we know that the current through C1 is also at a maximum at this time. We will plug in some actual numbers soon. R1 serves two purposes in this circuit: to limit the current passing through the circuit, and just as importantly, giving a means by which to *measure* the circuit current. Measuring the voltage drop across R1 gives us a means of doing just that.

Now, let's plug some actual numbers into the schematic above and see what happens. I've used *LTspice* in **Figure 2** to model the circuit. I've made R1 = 1000 Ω, C1 = 0.1 µF, and V1 a 10 V step function, equivalent to closing a switch. The resulting voltage at the junction of R1 and C1 is seen in **Figure 3**.

We have a new term, called the *time constant*, τ, to describe what's happening here. τ is simply the product RC. In basic units, τ is seconds, R is ohms, and C is farad. Using the values above, we come up with a time constant of 0.1 ms. But what actually happens at one time constant? Well, the capacitor isn't fully charged, obviously. In fact, it never gets fully charged. What happens at one time constant, is that the capacitor is at 0.632 times the supply voltage. Since our supply voltage in this example is 10 V, after one time constant, the voltage across the capacitor will be 6.32 V, as seen on the graph. Now, let's call each successive time constants, T1, T2, T3, T4, T5,..., etc. At time T2, the voltage will be 0.632 times that of T1; at T3, the voltage will be 0.632 times that at T2, and so on.

We can see from the graph that after five time constants (5 ms), the charge pretty much levels off. A capacitor is considered to be fully charged after five time constants, at which time the capacitor voltage is more than 99% of the supply voltage.

Half-life

Not too surprisingly, the system behaves similarly when a charged capacitor is being discharged through the same resistor, as shown in **Figure 4**.

As you can see, the capacitor never fully discharges, but drops in the same exponential manner in which it charged. In this case, after one time constant (0.1 ms), the voltage has decreased by 63.2%, or, more practically speaking, the voltage is 36.8% of the initial voltage, with the voltage at each successive time constant being 36.8% of the previous one.

If this entire scenario seems suspiciously like the radioactive decay half-life concept, you would be absolutely correct. Just as an exercise, let's temporarily re-define the time constant to be the time at which a capacitor discharges to precisely half of its initial voltage. How many such "new time constants" would you need for the capacitor to discharge to less than 1% of its initial voltage? As an additional "thinking point," what disadvantage, if any, would there be in using an RC "half-life" instead of the 63.2% or 36.8% "normal" time constant?

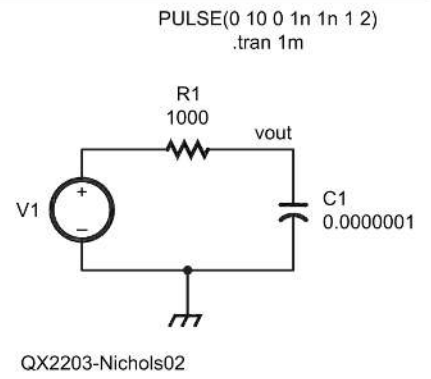


Figure 2 — Component values for the *LTspice* model.

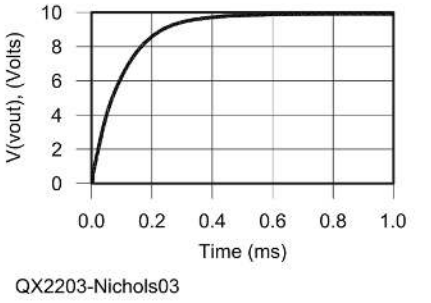


Figure 3 — Voltage on charging capacitor.

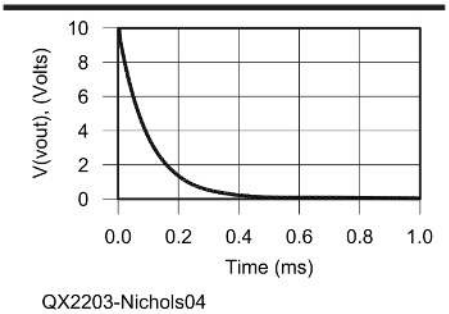


Figure 4 — Voltage on discharging capacitor.

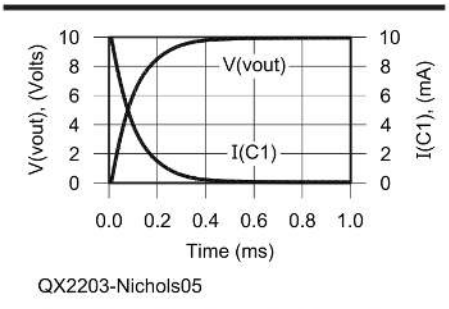


Figure 5 — Capacitor charging voltage (increasing trace) and charging current (deceasing trace).

Inductive

There's another time constant we should be aware of, but it's not quite so commonly used in electronic circuits, the L/R time constant. Like the RC time constant, the function is exponential in nature (as are most functions in the universe). In this case, the time constant (τ) is the inductance divided by resistance, again, using the fundamental units of seconds, henry and ohm.

An inductor doesn't hold a charge, per se; the magnetic field instantly collapses once a source of current is removed. This is why the formula is "upside down" in comparison with the RC time constant.

The charge and discharge curves of a series L and R circuit will be identical to the RC charge and discharge curves, except that the vertical axis will be current through the inductor, as compared to voltage across a capacitor in the RC case. L/R is seldom used as a timing circuit in practical electronics, whereas there are countless circuits that rely on RC time constants. However, the fact that such an L/R time constant exists speaks of the astonishing symmetry of the universe.

Reactionary

We have seen how capacitors and inductors respond or "react" to a change in applied voltage. And, there is no more drastic change one can contrive than an instantaneous application or removal of a voltage from a circuit component, whether an inductor, a capacitor, or a resistor. The response of an inductor or a capacitor to a change of applied instantaneous voltage is called, appropriately enough, reactance. As it turns out, the way a capacitor or inductor reacts to a sine wave voltage source is much simpler mathematically than how it responds to a sudden step-function voltage. Later on in this series, after establishing some calculus fundamentals — yes, we threatened this before, didn't we — as well as some ac principles, we will come back and revisit the RC time constant from a calculus perspective.

Putting on a good Phase

Let's go back to our original RC charging circuit, but plot both the capacitor voltage (rising trace) and the capacitor current (decreasing trace) of top of each other (**Figure 5**). Notice that the voltage and the current are out of phase, which is the main hallmark of a reactive component. In a

(Continued on page 28)

Brian Robert Callahan, AD2BA

19 Bank St., Troy, NY 12180; **callab5@rpi.edu**

DataTV – A Protocol for Embedding Data into SSTV Transmissions

Repurpose select portions of the transmitted image to carry other information.

Slow-scan television (SSTV) is a unique protocol among the amateur radio operator's toolkit. Going beyond mere words that CW and phone communications provide, SSTV permits the amateur operator the transmission of images that can be received and displayed by any station with an antenna and the software, often open source, necessary to decode such transmissions back into the original image.

Interestingly, SSTV has involved an implicit tradeoff: one may transmit images but those images are just that — images alone. Only what can be conveyed on the small television canvas may be communicated.

This tradeoff might appear obvious, since one of the earliest uses of SSTV was as a method for transmitting images from space **[1]**, **[2]**, **[3]**. But there is no technical reason why SSTV transmissions must be limited solely to images. With creative use of lessons learned from steganography and computer science combined with selecting quality off-the-shelf technologies for images and data, along with using some aspects of SSTV that are already available to amateur operators, arbitrary data can be embedded into SSTV images in transmission, and recovered by receiving stations.

In this article we will devise a new protocol, Data-embedded SSTV (DataTV), for embedding data into SSTV transmissions. DataTV extends the usefulness of SSTV by repurposing select portions of the transmitted image to other information. A DataTV transmission consists of two components: an image component, which is effectively an SSTV image, and a data component, which is additional data embedded in the image component using least-significant bit (LSB) steganography techniques.

The goals of this article are to demonstrate the feasibility of DataTV as well as provide a baseline model for how DataTV can be achieved, furthering the state of the art. Furthermore, this article serves as public documentation for a baseline DataTV protocol. Finally, this article serves as an introduction for how steganographic techniques could be applied to amateur radio transmissions, paving the way for future experimentation in this direction.

It is my hope that this article inspires further experimentation with protocol development and methods to embed additional data into amateur transmissions.

Prior art and considerations

Amateurs have experimented with steganographic techniques in SSTV before. Andreas Westfield discussed such a technique in a 2007 **[4]**. Westfield was primarily concerned with skirting a 2005 German Amateur Radio Ordinance prohibiting encrypted radio traffic at home. His system was able to embed up to 118 bytes of information. DataTV goes beyond Westfield's technique, as it can embed orders of magnitude more data in its data component.

At DEF CON 20, two amateur operators demonstrated how steganography can be used to send secret messages across many common amateur modes **[5]**. Citing 47 CFR § 97.113(a)(4) **[6]**, they caution early on in the presentation that their use of steganography in their presentation violates the rules on encoding messages for the purpose of obscuring their meaning.

However, I believe that steganography need not be a forbidden technique on the amateur airwaves. We just need to rethink its purpose. Rather than hide our use of steganography and using it to send secrets, we can instead announce its usage as a means of enhancing our transmissions for everyone, allowing everyone to do more and transmit more information with less overall bandwidth. This, I argue, is not dissimilar to the usage of cryptographic signatures in amateur transmissions, which is legal **[7]**.

Nonetheless, it would be prudent to seek FCC clarification prior to using open, announced, and non-obscuring steganographic techniques, and by extension DataTV. This prior art reviewed above demonstrates that it is known to increase the amount of information in a given signal. We need only to decouple its perception as a secretive and obscuring technique. I believe there is a positive argument to be made for

its approval in this form. I seek collaboration from other amateur operators to help create and push for a positive resolution for this clarification.

Finally, DataTV should be simple enough for the first-time amateur operator experimenting with protocol development to implement at least most parts of the system, with a good investment of time and energy. It should be able to be clearly understood and replicated across many skill levels of the amateur operator.

Selecting a DataTV image format

There exists prior art in the digitization of SSTV images that must also be explored. The primary technique in this space is known as Slow Scan Digital Video (SSDV), developed by Philip Heron with the UK High Altitude Society. SSDV is a method for digital transmission of SSTV wherein an SSTV image is converted to a specially crafted JPEG image, which is then transmitted as a series of packets **[8]**. For the purposes of DataTV, SSDV is important since alongside the image SSDV also transmits telemetry data, serving as evidence that it is feasible to transmit additional data with SSTV images, though the technique for transmitting this additional data differs greatly to that of DataTV. While SSDV claims that any digital mode that can carry text or data would be eligible to transmit SSDV, only an 8-bit RTTY implementation has been developed thus far.

Unfortunately, SSDV is a less than ideal choice for our DataTV protocol. As we will soon see, we will be using LSB steganography for embedding data. While it is possible to use LSB steganography with JPEG images **[9]**, nonetheless with JPEG being a lossy compression format, it remains a poor choice for LSB steganography using simple techniques **[10]**. Additionally, SSDV encodes its additional data outside the image itself, all of which is then wrapped in packets. In contrast, DataTV encodes the additional data inside the image.

Returning to our task of selecting an image format suitable for DataTV, PNG images provide a lossless method for image transfer. This is significantly better than JPEG because our embedded data will not be lost as a result of a lossy compression algorithm. PNG images are therefore a good choice for DataTV images. Another good choice would be 24-bit bitmap images, since these are uncompressed lossless images. Both PNG or 24-bit bitmap images should be supported as acceptable image formats for DataTV, with PNG preferred if only one image format is to be offered.

Unfortunately, not all SSTV programs accept PNG files. For example, the MMSSTV program for the Microsoft Windows operating system permits only JPEG and BMP images. This is the reason for accepting both PNG and 24-bit bitmap images in the DataTV protocol: this allows programs like MMSSTV to still be used to work with DataTV.

Selecting a DataTV data encoding format

The data component of DataTV transmissions has fewer overall considerations compared to the image component, as once a proper image component format is selected, we are simply dealing with binary data for the data component — simple ones and zeros. The contents of the data are not managed by the DataTV protocol, since that would defeat the purpose of encoding data into SSTV images.

However, there are a few considerations worth addressing and deciding upon for the data component. The first is the endianness of the data. All data should be in little endian format. The second is header information — metadata for the data component. The most minimal header should be 12 bytes in size and contain three copies of the same 32-bit number. That number should be the total size of the data component including the header. This is because the data component does not need to occupy all the space afforded to it. It is implementation-defined whether or not to provide a larger header with more information and/or additional copies of the header in the data component, for example, at the conclusion of the data payload. Larger and additional copies of the header reduce the total size remaining for the data payload.

Finally, the header should begin at the first pixel of the image component, or as close to it as possible, and there should be no gaps between the end of the header and the beginning of the payload.

Using least-significant bit steganography to embed data

Least-significant bit steganography is a common steganographic technique in which the least significant bit of each byte in a cover file, which can be for example an image or audio file, is used to store a message that can be reconstructed by concatenation of all such bits **[11]**. Nominally, the goal of any steganographic technique is to hide information in plain sight. DataTV flips this understanding of steganography on its head to repurpose the technique in a way that announces the embedded data rather than hiding it.

For DataTV, we do not care about the hiding of information; in fact, hiding information belies the very purpose of DataTV, which is to provide more data and information to receiving amateur stations. DataTV therefore by its very nature is an announcement that steganographic techniques have been used on the transmitted image. This in turn has the effect of turning steganography from a method of hiding secrets in plain sight into a public side channel in which additional information is transmitted — exactly what we want.

To understand the LSB steganography technique at work, we must first establish a few assumptions about what an image is for the DataTV protocol:

(1) An image is a 2-dimensional array, with X columns and Y rows, with the top left corner considered the origin: (0, 0).
(2) Each intersection of a column and row is a pixel.
(3) Each pixel contains one 24-bit value. This value is composed of three 8-bit values: one representing the color red, one representing the color green, and one representing the color blue.

Using LSB steganography, the least significant bit of each 8-bit color representation is used as one bit for the embedded data, yielding three bits of embedded information per pixel. In the worst case scenario, a pixel will have the least significant bit of each of its three 8-bit color representations altered. Even in this worst case scenario, the resulting color change should be imperceptible to the human eye. Though, as mentioned, hiding information is not a goal of DataTV, the fact that at worst the color change remains imperceptible to the human eye means that amateur operators using DataTV need not alter their source images in any way for use in DataTV transmissions, other than embedding the additional data. One can simply use the source image as-is and be sure that even with embedded data the visual quality will remain the same.

It is implementation-defined whether or not to offer the option to use more than one bit per color channel for the data component, so long as all color channels use the same number of bits and the option to use one bit per color channel remains available. For example, an implementation could offer the use of the least, second least, and third least significant bits per color channel. Additional bits increase the maximum theoretical size of the data component at the cost of potential

perceptibility of color change from the original image. It is up to the implementation and ultimately the amateur operator to decide if the benefit outweighs the cost.

Creating a DataTV transmission

Once the data to embed is selected and the image created in the image editing software of the amateur operator's choice, an LSB steganography utility must be used to combine the two together to create a DataTV image. There are many such utilities available online, many of which are open source. Which LSB steganography utility is used is immaterial to the DataTV protocol, modulo errors in the utility's implementation of LSB steganography. For future security, future articles and work will produce a common set of baseline utilities specifically for DataTV to accomplish encoding and decoding of the data component from the image component.

In this baseline specification, once a DataTV image has been produced, it should be transmitted via any data mode.

Considerations for theoretical data component limits

Future work will focus on improving robustness of the DataTV protocol so that it can be transmitted and recovered as a purely analog signal. When that is the case, specific considerations must be made for the existing constraints of analog SSTV transmission modes, as that dictates the maximum theoretical size of the data component. For this article, we will presume Scottie S1 as the selected SSTV method. According to Martin Bruchanov, OK2MNM, Scottie S1 has a resolution of 320 × 256 **[12]**, yielding a theoretical maximum of 30 KB of embedded data for a DataTV image transmitted via Scottie S1, calculated thusly:

320 × 256 = 81,920 total pixels per Scottie S1 image

81,920 × 3 bits of embedded data per pixel = 245,760 bits of embedded data

245,760 / 8 bits in a byte = 30,720 bytes

30,720 / 1024 bytes in a kilobyte = 30 KB.

Additionally, there are other techniques available for steganography over radio communication, which may be adaptable for analog transmission of DataTV **[13]**.

Receiving a DataTV transmission

After receipt of the DataTV transmission, LSB steganography software can be used to extract the data component and write it to the computer's file system. It would then be up to the operator what to do with the extracted data.

Error correction

As the embedded data component is potentially more sensitive to alteration than the image component, some level of error correction would be desirable for the embedded data component of a DataTV transmission. Below outlines what could be considered the minimum for error correction for the data component. Though it has some flaws, its ease of use makes it ideal for quick implementation. More robust DataTV implementations can and likely should provide improved error correction techniques alongside or as a replacement for this baseline error correction. SSDV, for example, uses Reed-Solomon codes allowing the receiver to correct up to 16 byte errors in a received packet **[8]**.

In his highly influential paper, Richard Hamming describes error detecting and error correcting codes **[14]** that can be used to provide some level of error correction. The simplest of which would be a repetition code, where each pixel uses all three of its bits of information to send the same bit three times. Upon receipt, majority logic wins: that is, whichever bit is represented at least twice is the one bit of information selected for that pixel. This is known as Hamming(3,1).

While it provides some level of error correction for DataTV transmissions, Hamming(3,1) suffers from a rather poor rate, since only one-third of all data component bits sent are actually used to convey the intended information. Even an upgrade to Hamming(7,4) would significantly improve the rate to four-sevenths, at the cost of additional complexity in encoding the data component and decoding software.

With the basic Hamming(3,1) error correction, the theoretical maximum data transmission falls to 10 KB. Not nearly as good as the original 30 KB of data with error correction omitted, but still capable of interesting data transmissions. As mentioned, an update to Hamming(7,4) would get significantly closer to the original theoretical maximum of 30 KB of embedded data, about 17.14 KB. As previously stated, Gains in the amount of information contained in the data component can be made by compressing the data component information prior to embedding.

Where this built-in error correction fails is when noisy channels prevent complete transmission or in the event that two or more data bits in a pixel happen to be flipped. To aid in completing receipt of DataTV transmissions over noisy channels, there exists external techniques which can be adapted, such as Packet Compressed Sensing Imaging (PCSI).

Further error correction with PCSI

A paper presented at *ARRL/TAPR DCC 2020* **[15]** describes a method for transmitting complete images where each individual receiving station may miss different parts of the transmission. A novel network between stations is created that allows all stations to cross-reference with the other stations their own missing parts of the transmission and offer to all other stations copies of the parts of the transmission it did receive and currently possesses. Together, as long as each part of the transmission was received by at least one station in the network, all stations through this system of cross-referencing and offering the parts they possess can all come to obtain a complete copy of the original transmission.

A PCSI network can be established in an ad hoc manner. Stations could then all receive a DataTV transmission and use the PCSI software to help ensure that all stations receive a complete copy of the transmission. This completed transmission could then be further processed by other SSTV software that supports DataTV. Since PCSI depends on multiple — ideally as many as possible — receiving stations, it is best used to augment error correction capabilities built into DataTV rather than as a replacement for such built-in capabilities.

Future work

There is one vitally missing aspect to this baseline DataTV protocol: transmissions in this baseline protocol are digital in nature. While this works, the DataTV protocol was conceived to be fully backwards-compatible with any RGB analog SSTV transmission mode, such as Scottie S1, not too dissimilar to how NTSC color television is backwards-compatible with NTSC black and white television.

Preliminary testing with Scottie S1 under ideal conditions suggests that while using this baseline DataTV protocol the data component of the DataTV transmission is detectable, it is not quite robust enough to fully recover the data in the data component. Once the robustness problem is solved a subsequent article will be submitted to *QEX* detailing the process. I welcome collaboration on this issue, since DataTV over analog transmission represents a significant step forward for amateur radio

and SSTV. It would allow amateur operators to ease into the new mode. DataTV over SSTV would permit all amateur operators to transmit and receive DataTV transmissions with the same SSTV software they are using today. The image component of a DataTV transmission would be by definition an SSTV transmission, which will always be accessible with today's SSTV software. Amateur operators could then simply extract the data component upon receipt.

Allowing DataTV to be transmitted over existing SSTV modes would go a long way towards ease of use and ease of adoption for DataTV, something that is unfortunately lost in this baseline protocol using data modes. Solving the robustness problem may require abandoning image-based steganography for the data component in favor of audio-based steganographic techniques.

Conclusion

This article describes and publicly documents DataTV, a new protocol for slow-scan television that provides amateur operators the ability to embed data into SSTV transmissions. This article also provides a rationale for the selection of image and data encoding formats suitable for embedding data using least-significant bit steganography.

It details how least-significant bit steganography is applied to a DataTV image to embed data, considers the theoretical maximums for the size of embedded data using the Scottie S1 transmission method. It also discusses rudimentary methods for error correction that can be built into DataTV as well as more sophisticated external error correction methods that can be used to augment the built-in error correction. This article ends on some points for future work, notably improving the robustness of the DataTV protocol so that DataTV can be implemented on top of existing SSTV formats, providing backwards-compatibility and increasing ease of use and ease of adoption of DataTV. This baseline explication of DataTV demonstrates the feasibility of embedding data into SSTV transmissions, furthering the state of the art.

Finally, this article provides an introduction to how steganographic techniques could be used in amateur radio transmissions, paving the way for future experimentation in this direction. It is my hope that this article inspires further experimentation with protocol development and methods to embed additional data into amateur transmissions.

Brian Callahan, AD2BA, holds an Amateur Extra class license. He earned his PhD at Rensselaer Polytechnic Institute and currently serves on the faculty of the Information Technology & Web Science program, also at RPI, where he conducts research into cyber security pedagogy and digital humanities, and teaches courses in cyber security and web science.

References

[1] NASA, 2015, "Luna 3 Mission Page." Available online: **https://nssdc.gsfc.nasa.gov/imgcat/html/mission_page/EM_Luna_3_page1.html**.

[2] S. Grahn, "MA-9 slow-scan TV experiment." Available online: **www.svengrahn.pp.se/radioind/Mercury/MercuryRadio.html**.

[3] A. Letten, 2010, "'Lost' Apollo 11 Moonwalk tapes restored." Available online: **https://web.archive.org/web/20140720193330/http://cosmos-magazine.com/news/lost-apollo-tapes-restored-and-broadcast/**.

[4] A. Westfield, 2007, "Steganography for Radio Amateurs—A DSSS Based Approach for Slow Scan Television." In *Information Hiding. IH 2006. Lecture Notes in Computer Science, vol. 4437*, J. L. Camenisch, C. S. Collberg, N. F. Johnson, and P. Sallee (eds.). Berlin: Springer.

[5] **https://media.defcon.org/DEF%20CON%2022/DEF%20CON%2022%20presentations/DEF%20CON%2022%20-%20Drapeau-Dukes-Steganography-in-Commonly-Used-HF-Radio-Protocols.pdf**.

[6] **https://www.law.cornell.edu/cfr/text/47/97.113**.

[7] **https://perens.com/2019/07/02/yes-it-is-legal-to-use-cryptographic-signature-on-amateur-radio-and-thats-important/**.

[8] P. Heron, 2016, "guides:ssdv [UKHAS Wiki]." Available online: **https://ukhas.org.uk/guides:ssdv**.

[9] J. Liu, Y. Tian, T. Han, C. Yang, and W. Liu, 2015. "LSB steganographic payload location for JPEG-decompressed images." Digital Signal Processing 38: 66-76.

[10] B. Sinha, 2015, "Comparison of PNG & JPEG Format for LSB Steganography." International Journal of Science and Research 4(4): 198-201.

[11] A. Cheddad, J. Condell, K. Curran, and P. Mc Kevitt. 2010. "Digital image steganography: Survey and analysis of current methods." Signal Processing 90: 727-752.

[12] M. Bruchanov, 2019, "Chapter 4: Formats of slow-scan TV transmission." In Image Communication on Short Waves. Available online: **https://www.sstv-handbook.com/download/sstv_04.pdf.**

[13] D. E. Robillard, 2013, "Adaptive Software Radio Steganography." arxiv:1304.7324. Available online: **https://arxiv.org/pdf/1304.7324.pdf**.

[14] R. Hamming, 1950, "Error Detecting and Error Correcting Codes." *The Bell System Technical Journal* 29(2): 147-160.

[15] S. Howard, G. Barthelmes, C. Ravasio, L. Huang, B. Poag, and V. Mannam, 2020, "Packet Compressed Sensing Imaging (PCSI): Robust Image Transmission over Noisy Channels." *ARRL/TAPR DCC Proceedings 2020*, Available online: **https://files.tapr.org/meetings/DCC_2020/2020DCC_Howard.pdf**.

Self-Paced Essays — #10 Almost AC

purely resistive component (like a resistor), the voltage and current will lie exactly on top of one another. Now the question probably comes to mind; how much are they out of phase? As it turns out, there's no way to answer this question, because phase only has meaning in association with radian frequency. As we move into genuine ac, we will find that both frequency and phase have meaning, and we can answer this question very precisely, and quite simply.

SPICE

This is a good time to encourage you to think about familiarizing yourself with *SPICE* circuit modeling, as we will be using it a great deal throughout this series. *The ARRL Handbook* has a great introduction to *SPICE* in its various flavors. For consistency, we will standardize with *LTspice*, which is also the ARRL default version. Furthermore, *LTspice* is free! We will teach you some really handy *SPICE* tricks along the way as we explore all the relevant electrical engineering principles.

Before our next session, break out your old trigonometry books and re-wobble some of those dormant brain follicles. As promised, everything you ignored in high school math will come back to haunt you in the next few essays. Enjoy!

Check Out What's New at DX Engineering!

X-Quad Antennas for 144-146 and 430-440 MHz

WiMo's lightweight and short-length X-Quad antennas come loaded with special features made with amateur operators in mind: switchable polarization plane (horizontal/vertical/circular right/circular left/diagonal); high gain compared to other antenna forms due to their stacked elements, boom length, and compact design; end-boom or mid-boom centered mounting capabilities; and usefulness in operating satellites. The X-Quad can be optimally deployed for the desired polarization by simply changing the feed. Remote Polarization Switches (sold separately) allow users to easily select planes on the X-Quad or other cross-polarized Yagis for maximum convenience and performance. **Enter "X-Quad" at DXEngineering.com.**

Telescopic Masts

For convenient portable operations, you can't go wrong with these strong yet lightweight aluminum telescopic masts. The masts' slotted aluminum tubing can be extended to the desired length and locked solidly in place with the attached clamps. Choose from models with maximum heights from 13 to 52 feet. These masts have a range of uses, including antenna construction, installation of weather stations, and mounting of a monitoring camera. Although primarily for temporary use, they can also be installed permanently with proper fastening to serve as a guying point for a dipole antenna or other functions. **Enter "WiMo Mast" at DXEngineering.com.**

Kelemen Trap Dipole Antennas

DX Engineering is excited to be the exclusive North American retailer of high-performance Kelemen Trap Dipole Antennas from WiMo. Choose from 47 models of high-efficiency, wide-bandwidth monoband or multiband antennas in a range of supported HF bands (160 through 10M), lengths, and power ratings (400W, 1,000W, or 2,000W PEP SSB and CW at resonance). These lightweight antennas come pre-assembled and ready to operate, with balun enclosed in a weatherproof polycarbonate case, clear PVC-insulated stranded copper wire, stainless wire clamps, and insulators. **Enter "Kelemen" at DXEngineering.com.**

ID-52A VHF/UHF Multi-Function D-STAR Handheld Transceiver

The HT you've been waiting for is here! ICOM's much-anticipated ID-52A comes with features and benefits designed to take your portable ops to the next level: 2.3" color display, a groundbreaking full-color waterfall band scope (the first of its kind for a VHF/UHF portable), an improved interface, an 80% increase in volume, integrated Bluetooth that allows for connectivity with a wide variety of external audio devices, the ability to send photos over the D-STAR network, DR mode for easy access to local repeaters based on internal GPS information, and loads more. **Enter "ID-52A" at DXEngineering.com.**

miniVNA PRO2 BT and Tiny Antenna Analyzers

Good things come in small packages, and these compact antenna analyzers prove it! With a frequency range of 1 to 3,000 MHz, the Tiny model (2.6" x 2.6" x 1.1") tests for SWR and impedance, and can be used for transmission measurements on band filters or amplifiers. As a vectorial analyzer, it can perform quadrupole measurements of S11 and S21 parameters, with the results displayed and stored as a Smith diagram. The PRO2 BT version (1 to 200 MHz) comes with a 1000mAh battery, providing four hours of full-scan operation, and Bluetooth interface—enhancements that allow you to use the analyzer directly at the base of the antenna. Other features include impedance range from 1 to 1,000 ohms; built-in battery charge; two separate RF outputs for I/Q for SDR experiments and IMD; and user-friendly interface for PC Windows, Linux, and Mac. **Enter "miniVNA" at DXEngineering.com.**

InnovAntennas

DX Engineering is pleased to be the exclusive North American dealer for finely crafted Yagis, rotatable dipoles, and log periodic antennas from InnovAntennas, each one the result of extensive testing and modeling to ensure stellar performance. Antennas are built with top-grade materials, including aircraft aluminum booms and elements, UV-resistant insulators, and stainless steel hardware. InnovAntennas' DXR models feature low maintenance; sturdy construction in a manageable size; no hairpins, traps, or matching links; and only one feedline for all bands. **Enter "InnovAntennas" at DXEngineering.com.**

ON ALL BANDS — ***DX Engineering's Amateur Radio Blog for New and Experienced Hams.***

Check Out DX Engineering's Facebook Page and YouTube Channel!

DX ENGINEERING®

Ordering (via phone) Country Code: +1
9 am to midnight ET, Monday-Friday
9 am to 5 pm ET, Weekends

Phone or e-mail Tech Support: 330-572-3200
9 am to 7 pm ET, Monday-Friday
9 am to 5 pm ET, Saturday

Ohio Curbside Pickup:
9 am to 8 pm ET, Monday-Saturday
9 am to 7 pm ET, Sunday

Nevada Curbside Pickup:
9 am to 7 pm PT, Monday-Sunday

800-777-0703 | DXEngineering.com

ON ALL BANDS YouTube

We're All Elmers Here! Ask us at: Elmer@DXEngineering.com
Email Support 24/7/365 at DXEngineering@DXEngineering.com

YAGI URBAN BEAM

The UrbanBeam is excellent for use in high density population areas or properties with small lot sizes, where a full-sized Yagi may not be an option. The distinctive shape and small footprint (15.5 sq ft turning radius) of the UrbanBeam helps make neighbors and spouses happier, while still delivering the exceptional results you would expect of a SteppIR Yagi. The UrbanBeam is a high-performance, two element Yagi on 20m-6m and folded dipole on 40-30m. With features such as 180 degree direction change, bi-directional mode and full element retraction for stormy weather. You can enjoy all the features of a SteppIR Yagi while chasing low-sunspot-cycle DX or rag-chewing with your friends!

YAGI URBAN BEAM

GO SMALL

steppIR
COMMUNICATION SYSTEMS

DETAILS & ORDERING:

www.steppir.com

425-453-1910