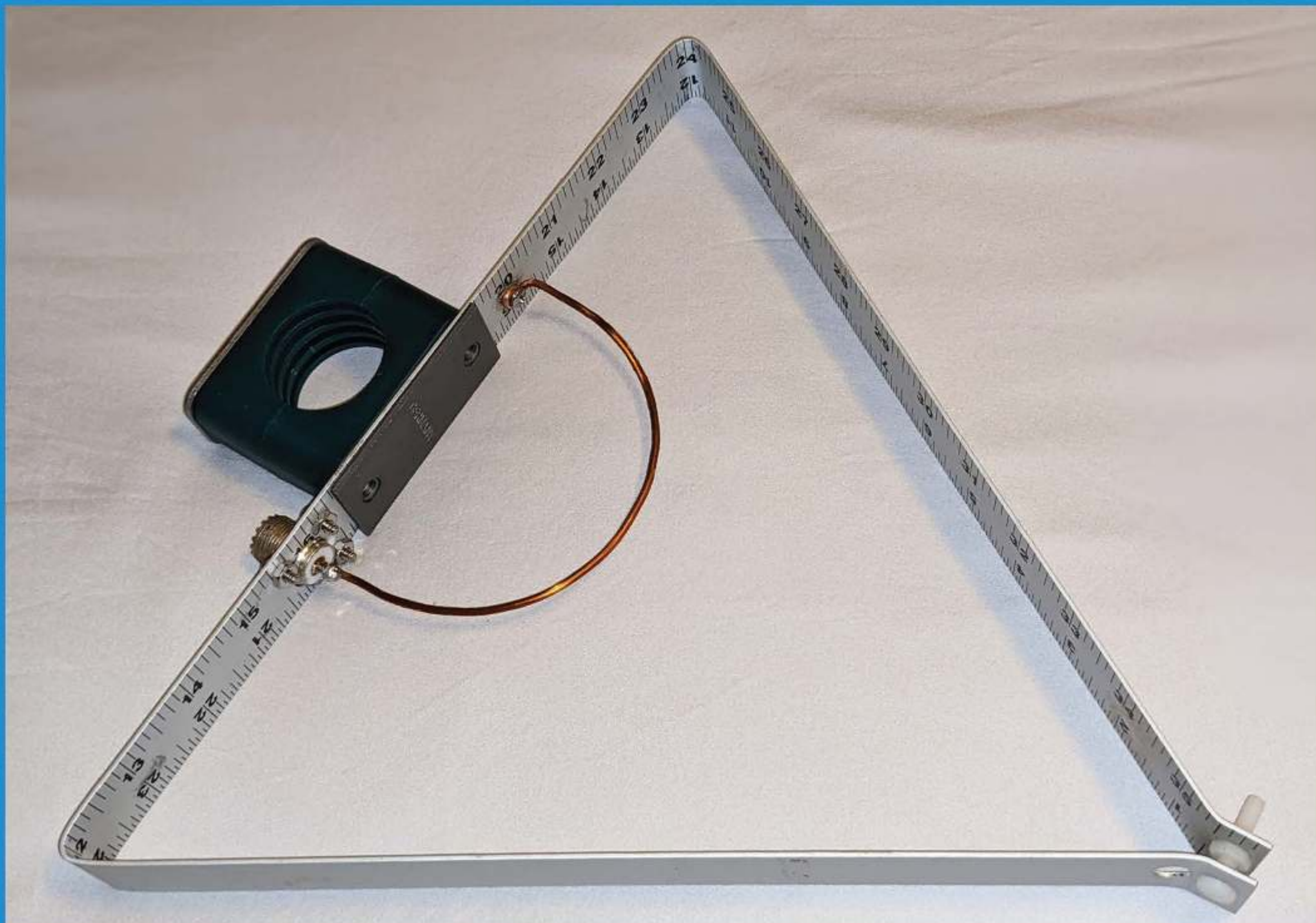
QEX

January/February 2023

www.arrl.org

A Forum for Communications Experimenters

Issue No. 336



W4RQ introduces an omnidirectional horizontally polarized loop for 2 meters.

The Legend Continues

TS-520

TS-570D/S

TS-590S

The TS-590SG

Back in 1973, Kenwood introduced the first affordable HF radio to the world, the legendary TS-520... 27 years later, the TS-570D and the TS-570S with 6 meters were by far the most popular HF and HF+6 transceivers on the market.

Be witness to the evolution of KENWOOD's pride and joy - the TS-590S HF transceiver - pushing performance and technology to its utmost limit, with the receiver configured to capitalize on roofing filter performance and IF AGC controlled through advanced DSP technology. Enter the TS-590SG. A new generation of high performance transceiver, with the type of high level response to meet DX'ers needs.

Don't be fooled by big boxes, high price tags, complex operation and broken promises. As Kenwood continues to build outstanding products with unparalleled performance and great value, it's no surprise Kenwood is rated as one of the leading choices for HF radios.

It's not too late to own an HF legend because we still build them today.

KENWOOD

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

www.kenwood.com/usa

ISO9001 Registered

ADS#16422

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. 335

Publisher
American Radio Relay League

Kazimierz "Kai" Siwiak, KE4PT
Editor

Lori Weinberg, KB1EIB
Assistant Editor

Ray Mack, W5IFS
Contributing Editors

Production Department

Becky R. Schoenfeld, W1BXY
Director of Publications and Editorial

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.

January/February 2023

About the Cover

Richard L. Quick, W4RQ, describes an easy to construct triangle loop antenna for use with horizontal polarization in the two-meter band. The loop radiation pattern is omnidirectional. This build is based on a 36-inch aluminum rule available at a hardware store. The loop fastens to a mast using a saddle clamp assembly. The omnidirectional pattern of the antenna has proven useful for a group of local hams who meet on the air in the SSB portion of the 2-meter band.

In This Issue

Index of Advertisers

Kazimierz "Kai" Siwiak, KE4PT

Perspectives

Digital Waveform Development

We repeat here the definition of the basic SDR system that we've touted in this column before. The basic SDR comprises some form of RF front end (a stable transceiver), followed by conversion between the analog and digital realms (a soundcard), along with a general purpose personal computer (PC). The SD or *software defined* part of the system is the amateur radio communications software that operates on the PC, producing a wide range of different communications protocols, or "waveforms" that usually are not native to the transceiver used as the RF front end.

To be sure, innovations in modern "proper" SDR receiver and transmitter platform architectures continue to migrate the boundary between the analog and digital realm closer to the antenna — but even those platforms benefit from the development of new PC-based waveforms or modes.

Hams continue to exploit the SD or *software defined* part of the SDR by adding new amateur radio communications software to the library of digital waveforms. Waveforms are usually tailored to address specific characteristics and needs, such as challenging propagation paths, or the general desire go make rapid fire minimalistic contacts with good link margin. SD waveforms and protocols breathe new life into older transceivers and enabled the enormous progress in digital waveform design.

Continue to watch these pages for the development of additional modulation waveforms, and for further SDR evolution.

In This Issue:

- Steven Davidson, K3FZT, designs and builds a Radio Message Server Winlink Gateway.
- Peter DeNeef, AE7PD, estimates diffracted fields inside a building near a window.
- Richard L. Quick, W4RQ, builds a horizontally-polarized triangular VHF loop.
- Eric Nichols, KL7AJ, in his Essay Series, explains filters.
- Brian R. Callahan, AD2BA, and Zhemin "Hisen" Zhang, KD2TAI, combine artificial intelligence and machine learning in a bot that transcribes heard audio into text.
- Lynn Hansen, KU7Q, reveals a unique method of constructing custom front panels.
- Steve Geers, KA8BUW, uses a microcontroller to build a CW audio filter.

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*, *OTA*, *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

Richard L. Quick, W4RQ
695 Fairway Dr., Plantation, FL 33317; **rich.w4rq@gmail.com**

Horizontally Polarized Two-Meter Triangle Loop Antenna

This omnidirectional horizontally polarized antenna is easy to build.

Several of my friends and I like to get on 2 meters SSB for a rag chew session each week. Some of us use the same vertical antenna we use for FM, while others use horizontal Yagi beams. You can imagine the chaos this causes, because the beams must be rotated often, and cross polarization adds to the dilemma.

We desired an inexpensive horizontal, omni-directional antenna that each of us could easily build. We investigated the halo, folded-dipole halo, the big-wheel, and the squalo antennas. Each of these have their good points, but would involve a complicated build, while others would be difficult to mount on a tower leg rather than the top of a mast. We found that these antennas are being sold at $65 to well over $150 on the internet.

The Triangle Loop

Randy Terrell, K9BCT, came up with a triangular halo using a gamma match feed, but we both found that a loop feed (**Figure 1**) works much better and is easier to implement. The result is this simple Triangle Loop Antenna made from the parts in **Table 1**.

Table 1
Materials for the Triangle Loop Antenna.

Qty.	*part*
1	36" aluminum rule (yardstick)
1	chassis mount UHF connector
4 ea.	#4-40 x ½" machine screws and nuts
9"	#12 AWG solid copper wire
1	#10-32 x ¾" machine screw and star nut
1	#10-32 x 1" Nylon machine screw
3	Nylon #10-32 nuts
1	Stauff saddle block clamp assembly.

The saddle block is the size of the mast or tower leg on which the antenna is to be mounted. It is available at Ace Hardware, Grainger, or eBay. Prices vary from $3 to $12 depending on source and mast size.

Randy used a Pittsburg 40" rule from Harbor Freight ($1.98), but I chose to use a 1-1/8" wide 36" rule from Ace Hardware ($6.59) because I broke my Harbor Freight rule when I tried to bend it too sharply in the square jaws of my vice. The Ace Hardware 36" rule is slightly thinner and is higher quality aluminum. I also clamped a round screwdriver in the vice with the rule, bending the rule around the screwdriver shaft so the bends are not too sharp. I learned my lesson!

The Build

Figure 2 shows the antenna dimensions. Drill a hole at the 16" mark, for the #10-32 machine screw that will secure the end of the feed loop wire.

Center the Saddle Block at 18" mark on the rule, then match-drill the rule for the mast Saddle Clamp assembly's mounting bolts.

Drill and mount the UHF connector at the 20" mark. Scrape the surface of the rule to make good contact and use antioxidant compound (for example Noalox).

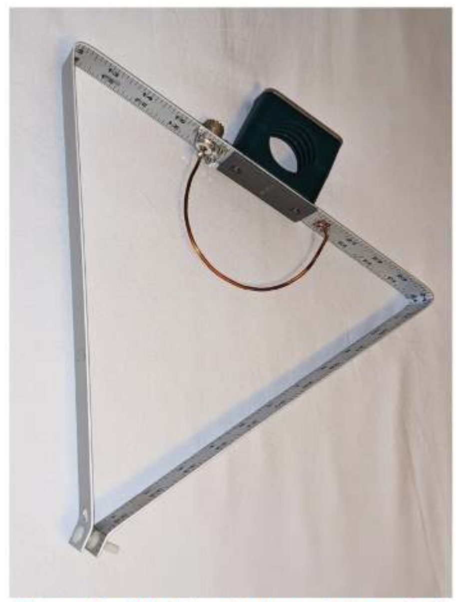


Figure 1 — The Triangle Loop Antenna fed by a secondary loop.

Match-drill the holes for the #440 screws that will hold the UHF connector on the rule.

Bend the first half inch, and last half inch of the 36" rule 60° in the same direction (that is, *up*).

The rule is then bent 120° in the opposite direction (that is, *down*) at the 12" mark and again at the 24" mark to form the triangle, with the two ½" ends parallel.

Drill a thru-hole in the ½" parallel ends and install the #10-32 Nylon screw and three Nylon nuts (see **Figure 1**). The Nylon screw and three nuts will be used to adjust the resonant frequency of the antenna.

Install the UHF connector using the 4 #4-40 screws and nuts.

Solder one end of the 9" wire to the center pin of the UHF connector, then form an 8" diameter half loop, using the extra 1" of the 9" wire to form an "eye" used to secure it to the rule with the #10-32 machine screw and star nut at the 16" mark, using Noalox after scraping the aluminum when securing the "eye" to the rule.

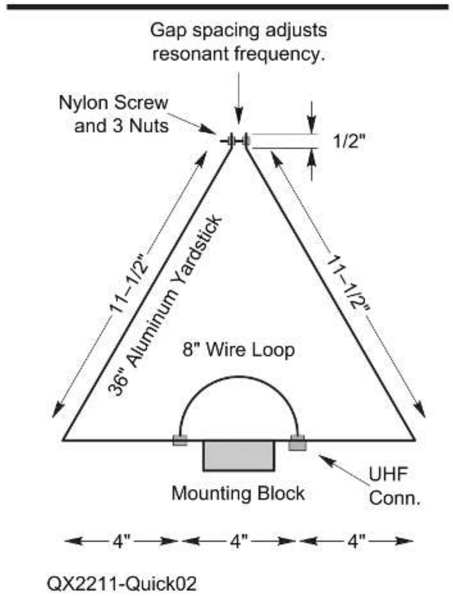


Figure 2 — The dimensions of the Triangle Loop Antenna.

Figure 3 — The Triangle Loop Antenna mounted on a mast.

Install the Saddle Clamp Assembly.

Mount the Triangle Loop Antenna on a mast or tower leg.

Set the resonant frequency with an antenna analyzer. This antenna is very forgiving, and is adjustable from below 139 MHz to above 148 MHz by adjusting the gap with the Nylon screw.

Results

The Triangle Loop Antenna mounted on a mast is seen in **Figure 3**. My cost to build the antenna was less than $10 plus the cost of the Saddle Clamp Assembly, and it took me less than an hour to build the antenna. The bandwidth is 2.16 MHz at the 2.15 SWR points. Resonance is 144.200 MHz, and the Return Loss is 36 dB. The impedance at resonance is 49.4 Ω resistive with a reactance of –1.43 Ω.

Testing the triangle Loop with a fellow ham who lives about 10 miles from me, I rotated the Triangle Antenna to point 0°, 90°, 180°, and 270° from his bearing and confirmed it is omnidirectional. I am making no claims about the gain of the triangle loop because I have not measured that, but 4*nec*2 modeling indicates the Triangle has a modest gain over a dipole.

I am anxiously looking forward to the time we all have these Triangle Loops in use, it will make it much easier for us to chat amongst the half dozen of us scattered around the county. This antenna is a very good addition to my station.

Stacking for Gain

Two of these triangle loops can be stacked, one above the other with a $\frac{5}{8}$ wavelength vertical spacing, and used with a phasing harness consisting of a UHF "**T**" connector, and two identical coax cables each having an electrical length of $\frac{5}{4}$ wavelengths (an odd number of $\frac{1}{4}$ wavelengths) at the resonant frequency. This will increase total omnidirectional gain by 3 dB.

Remember that an electrical wavelength in coax is shorter than the physical wavelength because of the cable velocity factor. Use a VNA or antenna analyzer capable of measuring the cable's electrical length. Cable manufacturer's stated velocity factor is accurate to only ±10%.

Richard Quick, W4RQ, is a retired electronics and metrology engineering technician and an Amateur Extra-class licensee, first licensed in 1977. He is a member of the ARRL, and is an avid CW enthusiast. Richard has co-authored the articles, "Does Your Antenna Need a Choke or Balun," March 2017 QST, *"Live Trees Affect Antenna Performance," February 2018* QST, *and "Small Gap Resonated Loop Antenna," September 2018* QST.

Steven Davidson, K3FZT
6925 Scotforth Rd., Philadelphia, PA 19119; **k3fzt@sjdmd.com**

New 2-m Winlink Gateway for Philadelphia Area Hams

An incremental and iterative process results in a working Radio Message Server Winlink Gateway.

In spring 2021 I was building an HF antenna and working on learning the essentials of emergency communication, and was growing increasingly frustrated because I was unable to connect to any Winlink gateway on VHF or HF. Sure, I could send email messages through Winlink using the built-in telnet facility, but the point was to use amateur radio RF to connect to a gateway and send email.

At the same time I was hearing from the Philadelphia County ARES Coordinator, Cliff Hotchkiss, KC3PGT, that both the ARRL and American Red Cross were focusing on Winlink as the preferred means of passing traffic in the event of an emergency. George Miller,W3GWM, the ARRL Eastern Pennsylvania (EPA) Section Manager, also had sent out a note confirming the ARRL requirement for Winlink usage in emergency communication (EmComm) [September 25, 2021 email to all EPA Emergency Coordinators].

Just as I thought I'd reached a dead-end I found a new direction from a fellow ham. As Field Day was wrapping up Jim Fisher, AJ3DI, suggested I consider standing up a 2-m RMS (Radio Message Server) Winlink Gateway. I was off and running. In this article I describe and show the work that went into bringing the idea of an RMS Winlink gateway to fruition.

What is an "RMS Winlink Gateway"?

I met John Galvin, N5TIM, through the Winlink email group **[1]** and he offered a draft of his "Using Winlink ... Configurations for Basic, Field and Emergency Communications Use with Diagrams" **[2]**. This invaluable guide helped me understand what I'd been reading about Winlink on the **winlink.org** authoritative website. It also helped me focus on what I was going to accomplish in the simple diagram illustrating an RMS Packet Gateway.

As with many projects I've undertaken in my career, this one started out with the idea and followed an incremental and iterative process to get to the final result of a working RMS Winlink Gateway. In what follows, I share a more structured process than actually occurred. My process was by turns fun, frustrating, fruitful, fulfilling and more expensive than necessary. By sharing the distillation of the work I undertook between early July and early December 2021, I hope you'll recognize the value of planning and documenting any ham radio project that requires more than a few minutes or hours of effort on one day.

Start by getting a notebook and pencil and dating your notes and ideas. You'll want to address at least the following components:

- Location – Where will you site the station and the antenna? Considerations include availability of power, connectivity to the house network and internet, spouse/partner approval of gear location and antenna appearance and location.
- RF Chain – A dedicated antenna, feed line, and radio are required for the RMS Gateway. The radio must have either a built-in soundcard or a 6-pin mini-DIN or alternate (DE-9 as on Alinco single band radios) for connection to an external sound card.
- Power – Sysop guidelines call for dedicated gateway equipment, preferably with automatic emergency power.
- Sound Card – Whether built-in or external, a quality sound card is required.
- Computer, operating system, and network — a 12 V dc mini-pc is sufficient and desirable as it will run off the same power supply and power backup used for the radio.
- Winlink Gateway registration and operating software — Authorization is required to operate as a Winlink Gateway.

In what follows I'll discuss each of these components and describe my choice and reasoning, illustrated by captioned photos.

Location

My shack, located in a corner of what was originally renovated as a home office, is a somewhat cramped arrangement. I was also looking for maximum height for an antenna. After discussions with my wife it was plain that she thought that anything behind the barn was preferable to anything at or close to the house. The back of the barn is the highest ground point on our property at 305 feet above sea level. There's power, 50 A split phase, but no network. I'll address both of these factors. The peak of the barn's metal roof is at ~20 feet and the peak runs approximately north-south. Inside the barn there was plenty of room to set up an operating position and install networking, power management, computer and communications equipment. This left me with the challenge of installing networking, but the tradeoff for the high point on the property seemed worthwhile. My friend, Clemens Sippel, a talented contractor, finish carpenter and former stage technical director with rigging experience suggested that an Ikea kitchen cabinet (40" H by 36" W by 14.75" D) could be mounted to the inside barn wall to provide shelves and an enclosure around the operating point. By leaving off the back of the cabinet during assembly and screwing 1 by 1 wood strips to the barn wall at the inside dimension of the cabinet, the cabinet could be screwed to the strips and well supported. See **Photo 1** and **Photo 2**.

RF Chain

Looking for a durable and reliable 2-m single-band radio I was repeatedly referred to Kenwood radios as the preferred choice. Recent and current models seemed rather expensive to dedicate to the gateway project.

I came across **hamprojects.info [3]** while investigating ways of setting up an Allstar node — I'm always looking for my next several projects — and that led to DINAH (DIN connector based Allstar interface for Hams) and that led to N1VG's page **[4]**, which started me on a search for a Kenwood TM-271A single band 2-m radio based on Kenwood's commercial radio line at the time of manufacture. It puts out 60 W at high power and has a big, finned heat sink. It doesn't have a 6-pin mini-DIN connector.

The European version of the radio, TM-271E, does have a 6-pin mini-DIN socket and what I learned from N1VG is that the TM-271A had solder pads where the connection to a mini-DIN could be made. So I followed the brief instructions and photos on the N1VG website and successfully installed a pigtail with a 6-pin mini-DIN socket to the radio. The radio has a single menu item to choose 1200 or 9600 bps, and since I'm running VARA FM in wide mode,

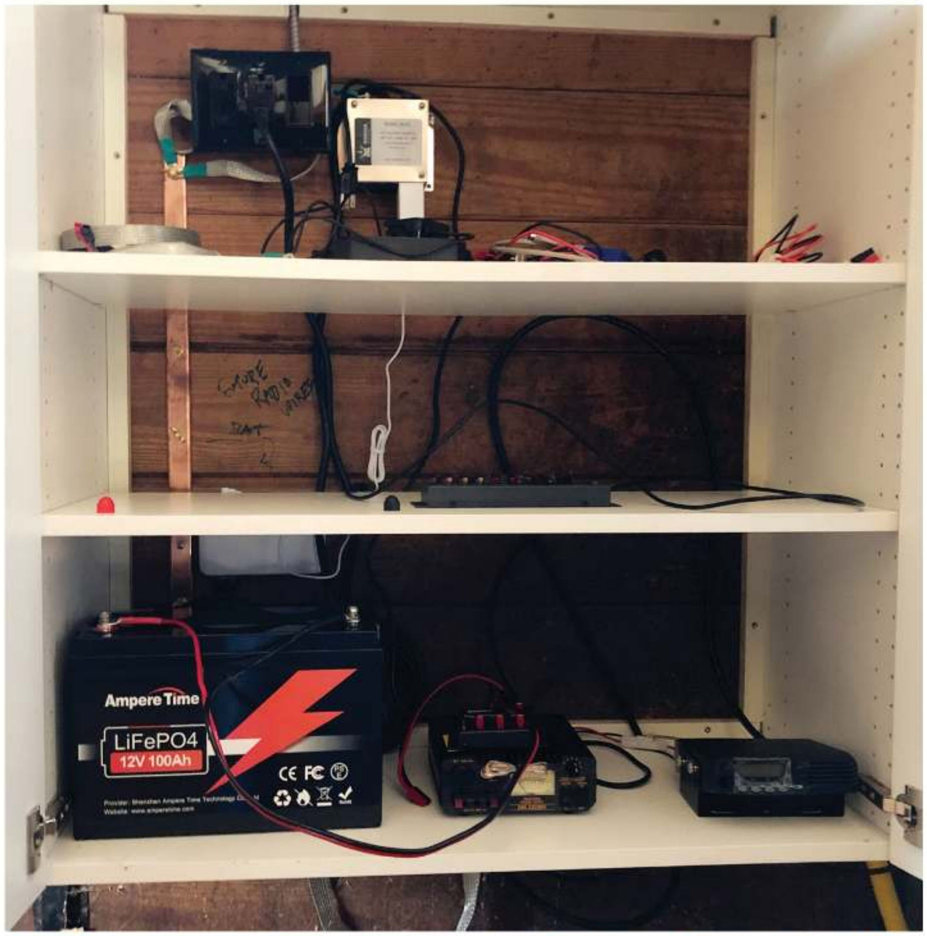


Photo 1 — Ikea cabinet houses power connection at the top; vertical ground bus at left, and provides space for the Winlink Gateway Station, backup power and future expansion.


Photo 2 — Detail of cabinet attachment to barn wall.

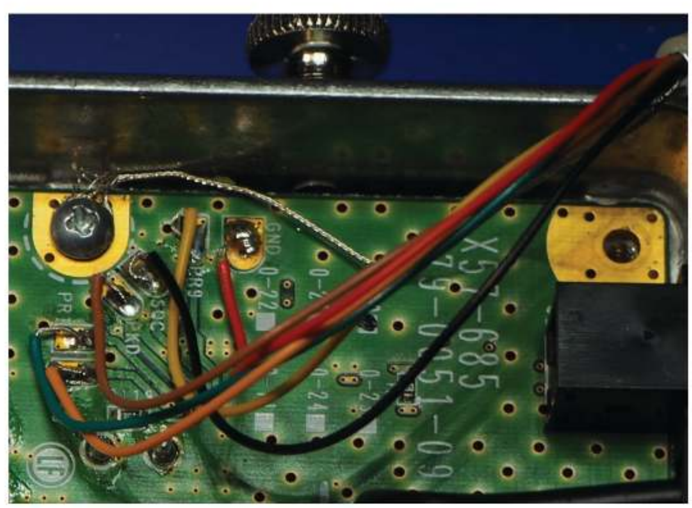


Photo 3 — Radio with the top off showing pads and wiring in place for the 6-pin-mini-DIN pigtail.

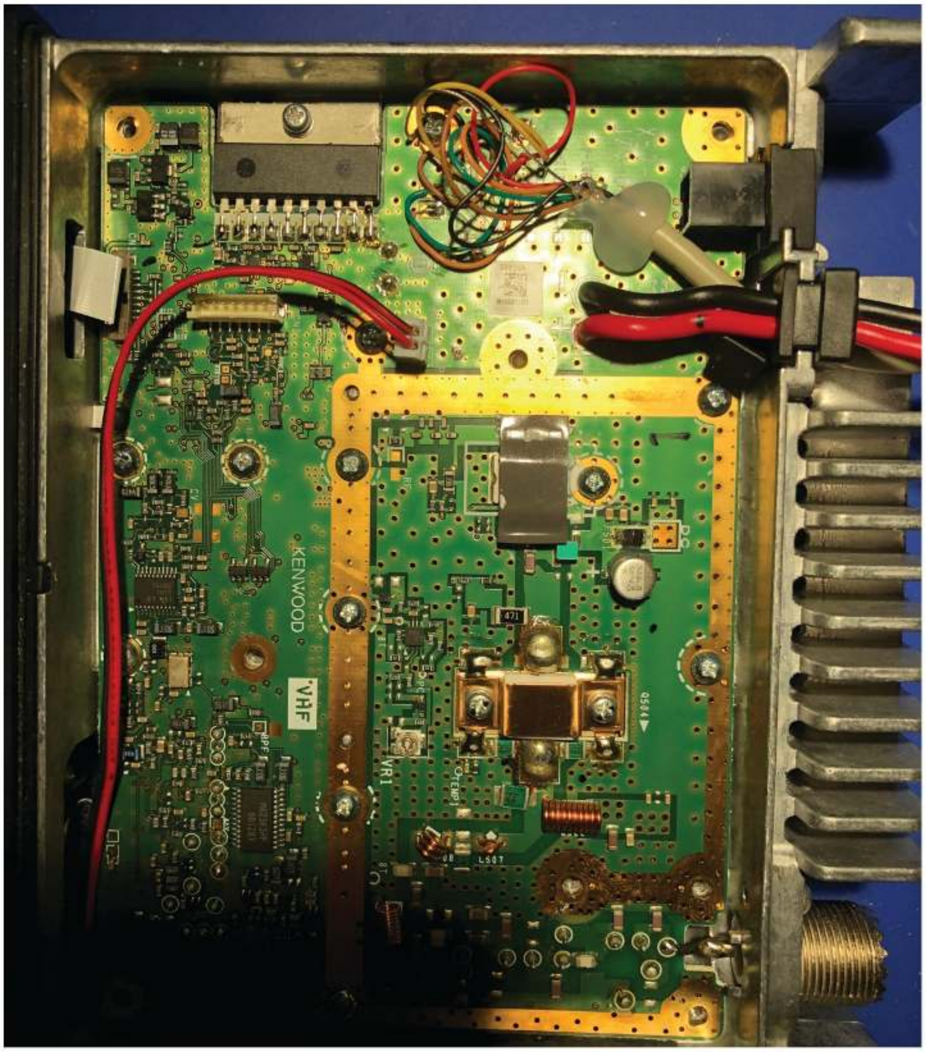


Photo 4 — Hot glue strain relief and wiring in final position just prior to reinstalling the cover.

Photo 5 — 2+ inch hole for conduit.

I set it to 9600. See **Photo 3 and Photo 4**.

Unbelievably, the QRZ swap meet forum came through the next day with a radio that reportedly had been boxed or sitting on a bench for most of the past decade. It was inexpensive and when received fired up easily putting out the full 60 W. I've since acquired, and modified, a second identical radio for warm backup courtesy of Barry Feierman, K3EUI, who was fund raising for Philadelphia ARES by deaccessioning old radios in return for a donation.

For the feed line, I used LMR-400 Ultraflex terminated with a Type-N connector at the antenna and a PL-259 for the Morgan M-303U coaxial lightning arrestor. The arrestor is mounted inside a KF7P MetalWerks **[5]** entrance panel box attached to an eight-foot ground rod that could not be sunk beyond 3 feet. So my friend, Clemens, dug an 8-foot long ditch about 18-inches deep and we laid the ground rod down in it. Deeper is better, but better and best can't always happen.

To get the feed line and ethernet cable from the mast mounted network station transceiver to the operating point inside the barn Clemens drilled a hole for the 2-inch conduit from the back of the entrance panel box into the barn. The conduit carries the bonded station grounding bus out to the entrance panel. The mast is also bonded to the entrance panel with 2-inch copper strap. See **Photo 5** and **Photo 6**. This medium size box can support future expansion to an HF RMS gateway and a remote wideband SDR and discone antenna.

I installed a Diamond X510HNA antenna mounted on a Rohn 50-foot (really 43.5-foot) tall push-up mast as suggested by Bill Popovik, W3AOK. The mast is bracketed to the barn at 5.5-feet and 16-feet above the ground and sits on an 8-inch square plate. The mast is bonded to the entrance panel with 2-inch copper strap. Steve Hoch, WU3I, led the crew that included Cliff Hotchkiss, KC3PGT, Clemens and me who together installed the mast and antenna. Also mounted to the mast, about 5 feet above the roofline was the station end dish for the Ubiquiti Building Bridge (UBB), which brings the house network out to the barn. The UBB ethernet connection is bonded to the mast where it is mounted. See **Photo 7**, **Photo 8** and **Photo 9**.

Power

Power in the barn was available at a panel in a space about 30-feet from where the antenna mast stands on the outside rear. Using 12-gauge BX armored cable I brought 120 V ac to the planned operating point on a dedicated 20 A circuit breaker. I had acquired an Alinco DM 330-MV switching power supply that would easily power the radio (13 A) and mini-pc (2 A) with headroom to spare.

I planned on using a 12 V battery for backup power, but wasn't sure how to manage the switching until a post-net QSO with Frank Rocap, N3FLL, that had been provoked by Jim Fisher, AJ3DI, talking about his Chinese copy of the device Frank described. Frank used a West Mountain Radio Epic PwrGate **[6]** to seamlessly switch power (up to 40 A) from ac powered supply to battery backup. That power management device, a 100 A Lithium Iron Phosphate (LiFePO4) battery from Ampere Time, and an Anderson Powerpole equipped electrically fused network controllable power-distribution panel completed the power setup. With the network controllable power a radio locked in transmit can be powered down without going on site. See **Photo 10**.

Sound Card

I built a Masters Communication DRA-70 from a kit although it's also available pre-assembled and tested. Kevin Custer, W3KKC, of Masters Communications offered his encouragement and availability to solve any problem I confronted while building the kit. The DRA-70 worked though it has since been replaced by a DRA-50 because VOX isn't required with Winlink running Vara FM and UZ7HO Soundmodem. The DRA-50 is the sound card in use for this gateway station running VARA FM in wide mode at 9600 bps and UZ7HO Soundmodem 1200 bps packet. See **Photo 11**.

Computer, Operating System and Network

I sought out 12 V dc powered computers and came across the Chuwi Herobox running an Intel "Gemini Lake" Celeron N4100, 8 GB of DDR4 RAM, and a 180 GB Intel SSD. This seemed to be plenty of power to run Winlink sysop software and TeamViewer, the preferred tool the moderators of the Winlink Google group use to support users. It has also run UZ7HO Soundmodem **[7]** for the 1200 bps packet Winlink gateway that shares the radio and computer. For those inclined toward experimentation and who are comfortable with minimal support, PiGate **[8]** offers an alternate RMS server.

The computer I acquired came with Windows 10 Home edition, and that's sufficient for the purpose. However, because I wanted to use Microsoft Remote Desktop (RDP) to connect to the computer in the barn from my office/ham shack computer I chose to upgrade to Windows 10 Professional. This was an unnecessary and time wasting exercise. TeamViewer has been more than sufficient for remote access and has no cost for private use on one computer at a time. For users who might need to use RDP for other reasons, purportedly RDP Wrapper Library **[9]** allows that use with Windows 10 Home edition. I have not tested that.

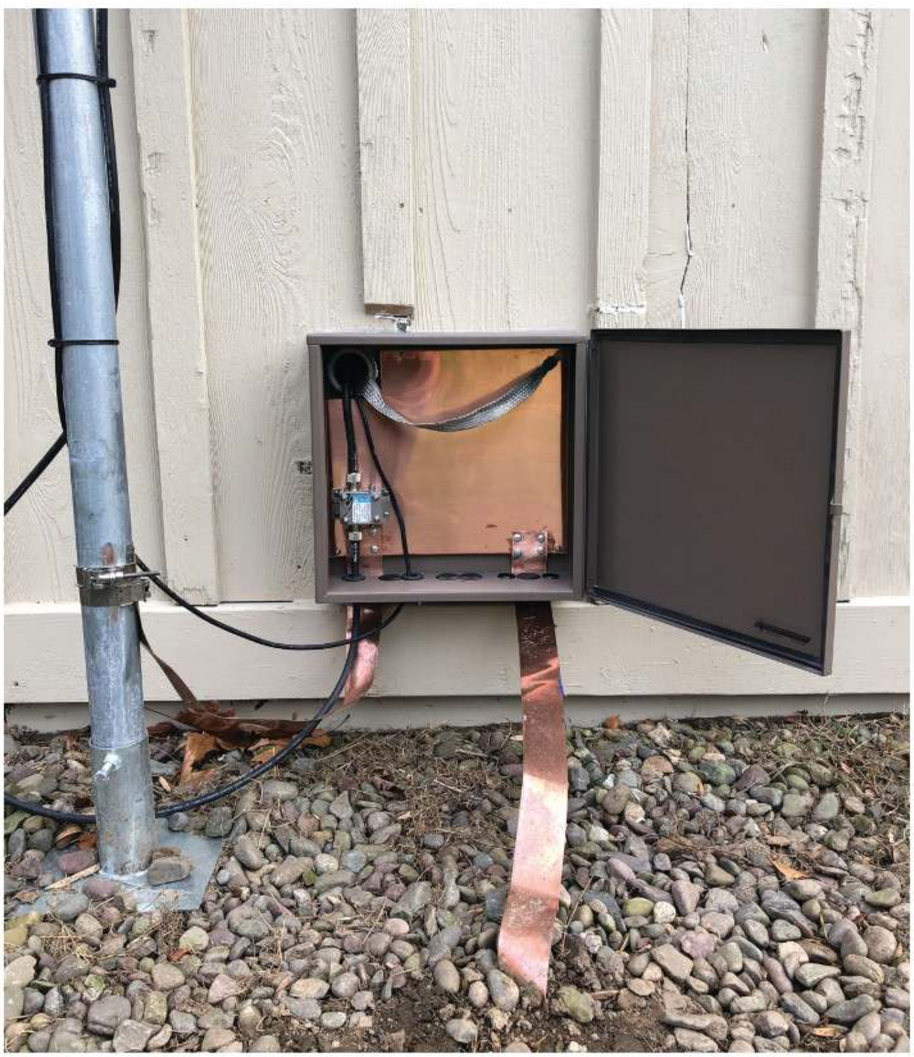


Photo 6 — Entrance panel box with coax lightning arrestor, ethernet from UBB to switch and tinned copper braid from station ground bar.

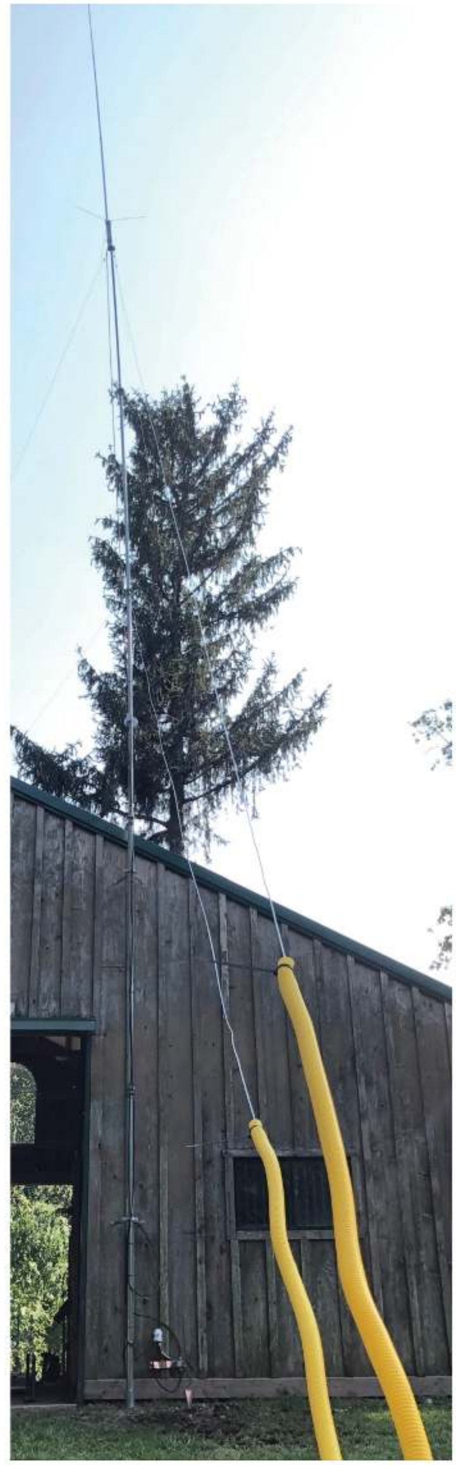


Photo 7 — "Antenna up!" For the first time.

Photo 8 — "Antenna going down" two weeks later.

Photo 9 — "Antenna back up" two hours later. The narrow barn hadn't allowed for enough spread for three point guys, so a fourth point at an approximately right angle to the barn peak trued up the mast.

Photo 10 — Epic PwrGate charges the battery and seamlessly powers the system if grid power is lost.

Getting network access from the house to the barn was relatively easy if a bit pricey. My house network is built upon the Ubiquiti Unifi system. The Ubiquiti Building Bridge (UBB) was readily available and after getting through an 18-inch-thick stone wall to get the cable outside and updating the firmware on both 6-inch diameter dish equipped 60 GHz transceivers the connection to the barn was installed. Tip: update software while both dishes are on your bench. An eight-port switch supporting power over ethernet (PoE) and a PoE powered access point completed the installation in the barn. This supports direct connection over ethernet for the computer, the West Mountain Radio Rigrunner 4005i **[10]** wireless connection for family and visitors in the area of the barn and connections for the two floodlight cameras I installed front and rear on the barn. See **Photo 12**.

My house network is served primarily by Verizon Fios, but in preparation for the project and in the interest of reliability, I had Comcast internet installed for fail-over support. The house network is configured with PoE for the access points, and the entire network is powered from the transfer switch panel providing grid power or natural gas fired 20 kW generator power that the previous owners installed.

Winlink Gateway Registration, Operating Software and Operating Frequency

I had registered with Winlink early in 2021 and had sent messages both for testing purposes and in response to the EmComm Technical Group's weekly Winlink Thursday exercises. Registration as a gateway required evidence of prior use of Winlink and agreement to the "Sysop Guidelines" **[11]**, **[12]** which contains recipes and advice for configuring Winlink Express (client) software on many different radios and sound cards. It also — in conjunction with the help files in the three main Winlink Sysop programs (RMS Packet, RMS Relay, and RMS Trimode) — offers plenty of information for configuring a Winlink gateway. The details are mostly there, supplemented by excellent support by the list owners/moderators of the Winlink programs group **[13]**.

Historically, Winlink gateways supported connections using a variety of ham radio digital modes. Likely packet and Pactor are among the best known and most widely

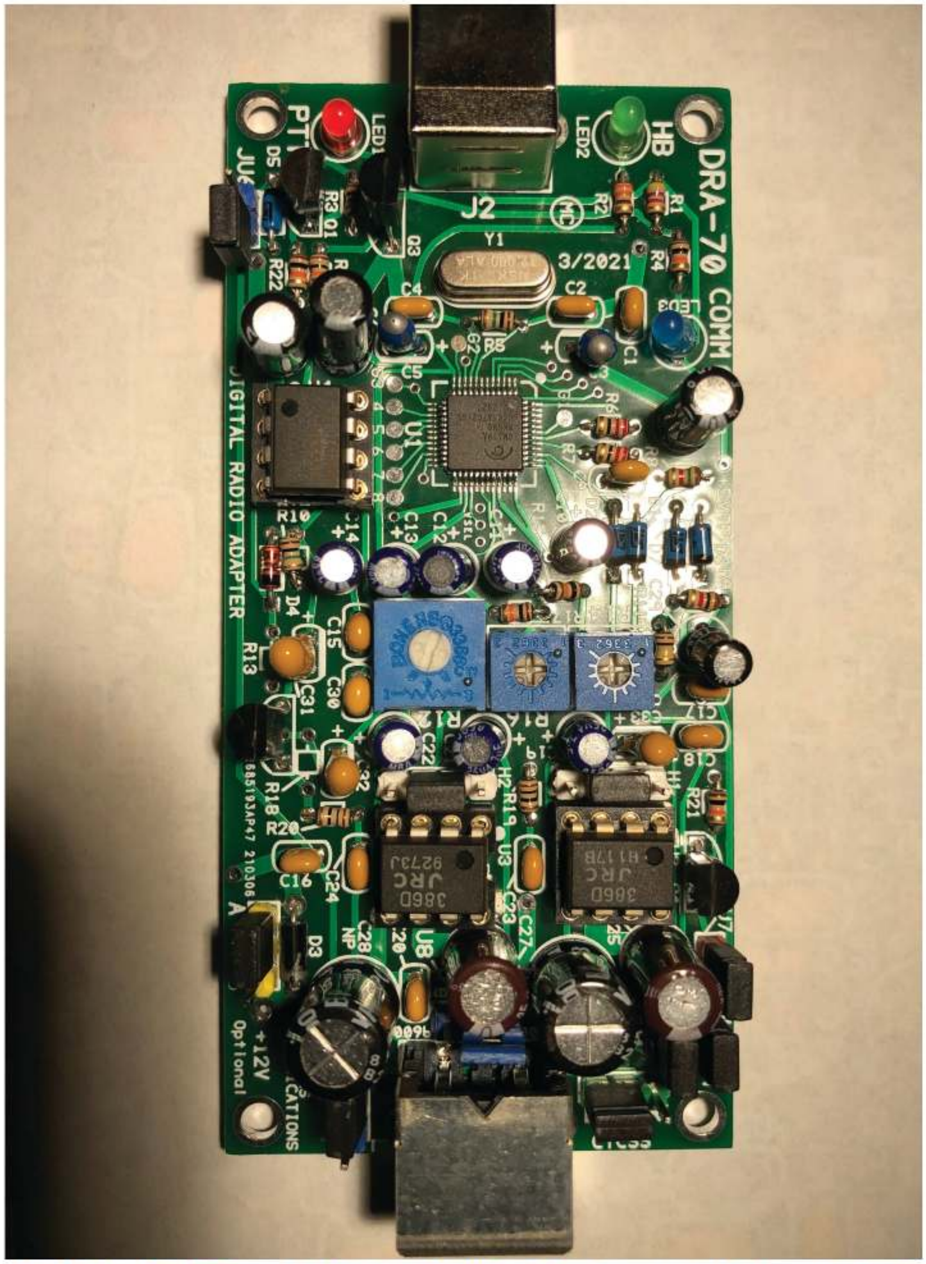


Photo 11 — DRA-70 completed, includes USB type B port connection for computer and a 6-pin mini-DIN connection.

Photo 12 — Transceiver dish and ethernet lightning arrestor on the house.

used though the Winlink folks themselves support Windmor (now deprecated) and Ardop. Of late, VARA FM on VHF and VARA HF, software modems **[14]** from José Alberto Nieto Ros, EA5HVK, have become widely used. I implemented VARA FM at 9600 bps in Wide mode and 1200 bps packet. RMS Packet Dual-Mode (Packet and Vara FM) setup using a single radio and sound card by Scott Curie, NS7C, **[15]** was an invaluable aid in setting up and configuring the sysop software. Scott came through when contacted directly by email with critical troubleshooting assistance. Also useful was a packet centric document I found **[16]**. Both of these documents come from the Winlink site, a rich resource.

The last step was picking a frequency for operation. I relied upon the Area Repeater Coordinating Council Band Plan **[17]** for 2-m and selected the sub-band of 145.510 – 145.790 MHz reserved to digital and experimental modes (20 kHz spacing) by the ARCC band plan despite the regular appearance of FM analog QSOs on this and adjacent frequencies. K3FZT-10 is now on the air and serving hams throughout the Delaware Valley at 145.670 MHz. See **Photo 13**.

Photo 13 — The completed station with room for storage and expansion.

Acknowledgements

I've mentioned many hams and vendors in the text, but particularly want to acknowledge my friend Clemens Sippel a craftsperson, rigger and problem solver nonpareil. In addition to general encouragement Cliff Hotchkiss, KC3PGT, Philadelphia County ARES Coordinator, helped put up the antenna. Jay Silber, WA2UAR, helped test the gateway and provoked me to do a better job adjusting the audio levels. Perry Pepper, KC3JUD, helped test the gateway. Along with me, these three Yaesu FT-991A owners all suffered through our learning curve of digital sound card modes under the tutelage and with the encouragement of Barry Feierman, K3EUI. Ron Wenig, NY3J, and Glenn Allison, N3MEL, provided a wide range of advice particularly for UZ7HO Soundmodem packet implementation. Daniel Wagaman, W4GMN, editor of the Phil-Mont Mobile Radio Club newsletter *The Blurb* reviewed, commented upon and generally assisted the improvement of the manuscript. "It takes the whole community."

Steven J. Davidson ("Steve"), K3FZT, holds an Amateur Extra class license. He was first licensed as KN3FZT, then K3FZT at age 13 in January 1964 after a Novice and Technician class given at the Franklin Institute in Philadelphia, and co-sponsored and taught by then Philmont Mobile Radio Club (PMRC) members. Now he's a PMRC member. A retired emergency physician and medical informaticist and former Philadelphia Fire Department Medical Director, he naturally gravitated to digital modes and ARES and EmComm. He serves as a VE for both the ARRL and GLARG. In addition to the Winlink Gateway in the barn there's now a WIRES-X node for one of the MARC UHF repeaters and he's building out a packet BBS and gateway for the Mid-Atlantic packet network. AREDN connectivity is in his sights. He prefers projects to operating, but when he operates it's mostly FT8 and rag chews on 160 and 75 m recapitulating the late nights when as a youth he listened on his Heathkit shortwave radio to many an old timer. He also operates the digital voice modes of DMR and Fusion/WIRES-X and maintains an Allstar node.

Notes

[1] **https://groups.google.com/g/winlink-programs-group/about**.

[2] **https://www.qsl.net/n5tim/DOCS/Winlink_Configs_2021_V2.pdf**.

[3] **https://hamprojects.info/**.

[4] **http://n1vg.net/**.

[5] **https://www.kf7p.com/KF7P/EntrancePanels.html**.

[6] **http://www.westmountainradio.com/product_info.php?products_id=epic-pwrgate**.

[7] **http://uz7.ho.ua/packetradio.htm**.

[8] **http://www.pigate.net/**.

[9] **https://github.com/stascorp/rdpwrap/blob/master/README.md**.

[10] **http://www.westmountainradio.com/product_info.php?products_id=rr_4005i**.

[11] **https://winlink.org/content/join_gateway_sysop_team_sysop_guidelines**.

[12] **https://winlink.org/content/winlink_book_knowledge**.

[13] **https://groups.google.com/g/winlink-programs-group**.

[14] **https://rosmodem.wordpress.com/**.

[15] **https://www.winlink.org/sites/default/files/RMSE_FORMS/rms_packet_dual_mode_single_radio_setup_0.pdf**.

[16] **https://winlink.org/sites/default/files/RMSE_FORMS/quick_setup_guide_for_winlink_sound_card_packet_for_vhf-uhf_on_windows_v1.2.pdf**.

[17] **http://www.arcc-inc.org/_files/ugd/ed9285_9a3724a1da3b402c88d7825cbd5058dc.pdf**.

Lynn Hansen, KU7Q
10914 Westchester Rd., Lava Hot Springs, ID 83246; **lynn@lynovation.com**

Custom Front Panels with a Twist

The author reveals a unique method of designing and building very low cost custom front panels.

Anyone who builds their own equipment can tell you that packaging their project is by far the most challenging part of the process. Adding requirements that the finished product must look good and function well compounds the challenge exponentially. A quick search of the ARRL archives for front panel construction ideas reveals that this subject has generated quite a few articles over the years.

In this article I'm going to review the process that I came up with to create a professional looking custom front panel for my latest project **[1]**, see **Figure 1**. Not only does the front panel look professional, it is extremely inexpensive, costing around $2.00 in quantities of 5. In addition, it includes a built-in cable management system to connect the controls on the front panel to the main circuit board, meaning no more rat's nest of wires between boards!

The Catch

About now, you're probably wondering, "what's the catch?" There's always a catch and this method is no exception. The catch is; you may need to learn something new! You see, the front panel on my SO2R controller is a printed circuit board (PCB) and to create a similar panel for your project may require you to learn how to run a schematic capture and PCB layout program. If you already have a program that does this, and allows you to create custom footprints (the component layout patterns for the PCB), you're ready to start. If not, I highly suggest you look into Kicad **[2]**, a free schematic capture/PCB layout program that runs on just about any operating system out there.

I'll be using Kicad throughout this article to demonstrate my techniques. Installing and running Kicad is outside the scope of this article but there are plenty of online resources available to teach you everything you need to know about the program.

My Front Panel Overview

As mentioned, the front panel on my project is a PCB. The top layer of the PCB is blank — the copper layer is not etched and is at ground potential. It has a black solder mask and white silkscreen lettering. You can choose from several solder mask colors for your project. The silkscreen lettering is pretty rugged and should wear well. You can get the silkscreen in any color you want… as long as it's white — or black on a board with

Figure 1 — Completed CTR2-Mini SO2R Controller featuring the PCB front panel created using the techniques described in this article.

a white solder mask. A close up of the front panel is shown in **Figure 2**.

Make sure the front copper layer on your board is connected to the bottom copper layer with through plated holes or vias **[3]**. If it is not, the board house will build the board without a top copper layer or solder mask, meaning you're front panel is going to be a pretty shade of raw fiberglass with white silkscreen labels.

One cravat with using Kicad is that the font used for the silkscreen text is fixed and can't be changed. You can resize it. If you really want to use another font you can generate your labels in another program then import them as footprint files. For control panel lettering I find the default font perfectly fine.

The controls and LEDs mount to the bottom of the PCB. The bottom copper layer contains pads and traces that connect the leads from each control to other controls or to the 0.1" pin headers that connect this board to the lower board. On my project, inexpensive 8" DuPont female/female jumpers provide the 24-conductor wiring loom between boards.

I designed custom footprints that provide soldering pads for each control and LED. These will be discussed later on. Short jumper wires connect the terminals on each control to the pads on the PCB as shown in **Figure 3**. You can use insulated stranded wire like I did here or you can use bare #26 AWG solid wire from CAT5 cable to make these connections.

One of the requirements of this design is that no through-hole component leads are allowed to spoil the front face of the panel. This necessitated using SMT resistors and capacitors. Two 12-pin SMT 0.1" pin headers were also used in place of normal through-hole pin headers. **Figure 4** shows an SMT header and an LED before soldering. To solder the header to the board just apply a small dab of solder to one pad. Position the header and reheat the solder on that pad. Once you're happy with the position of the header, apply solder to the rest of the pads. To install the LEDs, just bend their leads at a 90° angle and trim off the excess lead from each side. Drop it into the hole on the panel and solder it to the pads. Pay attention to the anode and cathode lead location before you trim them so you don't install the LEDs backwards.

If you're unsure about working with SMT components most board houses will provide SMT components and assembly service for a small charge.

Because this is a strict single-sided PCB, careful layout was required to eliminate all crossover traces. I found it beneficial to layout the signals on the pin headers in the order of the controls they connect to so unobstructed runs could be used. Once this board was laid out, the pin assignments on the header were used to design the lower board. That board is double-sided so it's much easier to route traces over each other.

The most tedious part of assembling this board is cutting, tinning, and installing the 1" jumpers. But even this is way easier than creating a point-to-point wiring loom to individually connect each control to the bottom board.

Let's Get Started

Packaging your project should be part of your initial design effort. While you're brainstorming your latest gee whizzer gizmo think about how you — or another user — will use it. What kind of user interface will it

Figure 2 — Close up of the completed front panel PCB.

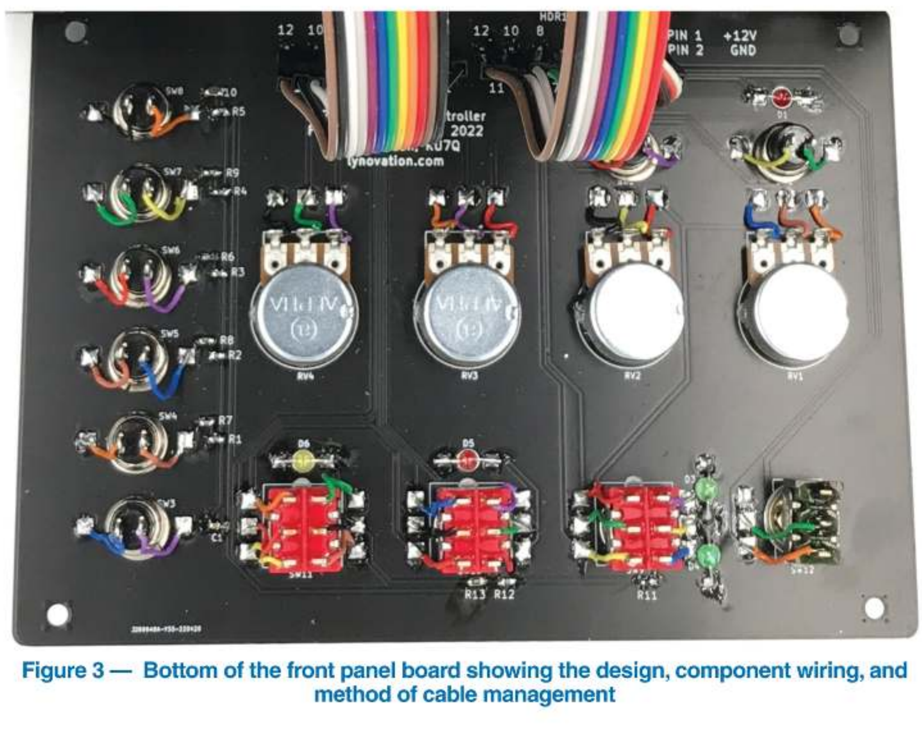


Figure 3 — Bottom of the front panel board showing the design, component wiring, and method of cable management

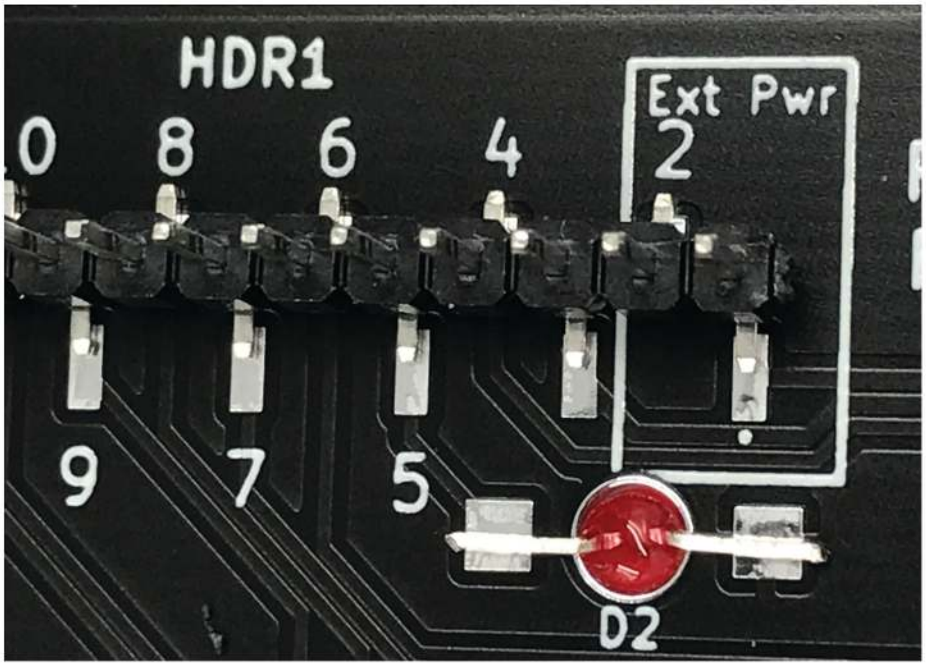


Figure 4 — Close up of an SMT .1" pin header and LED before soldering.

have and where will the controls be located? Will they be mounted on a vertical panel, a horizontal panel, or a sloped panel? Perhaps you're duplicating a project found in *QST*. In that case you already have an idea of how you want to lay out the front panel based on how the original appears in the article. But there's no reason why you can't add your own touches to the design and rearrange the controls to fit your operating style.

With the physical layout in mind, draw up a sketch of where the controls will be mounted on the panel. Make this drawing as close to scale as you can. This not only gives you an idea of how large your panel needs to be but it gives you a chance to organize your controls in a logical order. Group your controls by function. My controller has four control zones, the Tx Enable zone, the gain adjustment zone, the switching zone, and the function button zone.

While you're working on the panel layout, pay special attention to each control's location, spacing and alignment. Try to keep them balanced so the panel doesn't feel heavy on one side and light on the other. For example, the controls on my controller are laid out in a three-row, five column matrix with equal distances between rows and columns originating from the center of the board. The function buttons on the right side don't line up with the other rows but they do line up with the vertical columns. In my panel design everything has balance, a place, and a purpose. The more time you spend on this aspect of your project the happier you will be with the final result.

Next, start looking for a suitable enclosure. For my project I determined a panel about 6" by 4" was needed to mount 16 controls and 6 LEDs. Supply outlets do an outstanding job of linking manufacturer datasheets to the products in their database. Search through their listings until you find an enclosure that fits your needs. For my project I settled on the PacTec KEU-7 sloped panel **[4]**. According to the datasheet **[5]** this enclosure has a blank front panel insert that is 5.876" by 4.125" with four mounting holes 5.3" and 3.735" apart. Perfect, now I have the dimensions for my PCB.

Let the Fun Begin!

The first thing you'll need to do in Kicad is create a schematic for your front panel. If your project is simple you may be able to locate all of its components on the back of the front panel, but normally your front panel will be separate from the main board and require a wiring loom to connect the two. In this case you'll need two schematics: one for the front panel and one for the main board.

To create your schematic, open Kicad and create a new project. Once created, click the Schematic Layout Editor icon to open Eeschema, Kicad's schematic editor. You'll need some symbol libraries to do anything useful so click the Preferences… Manage Symbol Libraries icon to load what you need. Kicad comes with a huge symbol library and you can also download additional symbols from part suppliers. You can use the generic symbols from the Kicad library to create your schematic then assign custom footprints to each symbol for the actual board layout.

Kicad includes a schematic symbol editor. This editor allows you to create your own symbols or rearrange existing symbols to make new devices. For instance, I need a 3PDP switch for this project. There wasn't a symbol for this switch in the Kicad Switch library so I took an existing DPDT switch and added another pole to it. You can find videos online to learn the basics of editing symbols.

Try to keep your schematic organized, clean, and easy to follow. I try to minimize the use of the global signal labels to make the circuit easier to follow. That said there are times where signal labels help declutter the drawing.

The schematic captures the electrical functions of your device. Once you're satisfied with it, it's time to move on assigning footprints.

Footprints

Footprints are files that describe the physical attributes of the schematic symbol they are assigned to. A footprint must be assigned to every symbol on your drawing. The pin number on the footprint correlates to the pin numbers on the schematic symbol. Fortunately, Kicad comes with a huge footprint library too so you don't have to reinvent the wheel every time you add a new device. Part suppliers are also doing an outstanding job providing footprints along with the schematic symbols for the components they sell. However, when designing a front panel PCB you won't be using 'normal' footprints, you'll need to use footprints I have developed, or create your own. They're not hard to create and the Internet is full of helpful videos explaining how to create custom footprints. I'll cover the basics in the next section.

Creating Your Own Footprints

For this project I needed custom footprints for the pushbutton switches, LEDs, toggle switches, and potentiometers. My footprints consist of a mounting hole for the component and pads to wire the components terminals to.

In Kicad there are several layers, or

attributes, for each footprint object. **Figure 5** shows my potentiometer footprint in the editor. Download the datasheets for the components you will be using to get the dimensions you need for your footprints.

In the potentiometer's footprint the yellow circle in the center defines the hole for the shaft that will be drilled in the PCB. The datasheet **[6]** shows the bushing as being 6.35 mm (¼") in diameter. I made the diameter of this footprint object 7.2 mm and the hole in the middle of it 7 mm. This leaves about 0.1 mm of exposed copper around the hole and added an extra 0.65 mm to the hole diameter to allow for the plating that will be applied to the inside of the hole when the PCB is built.

The gray circle is on the Front Fabrication layer. This circle is 20 mm in diameter and represents the size of the knob I planned to use on each pot. This layer does not appear on the PCB itself but it helps me determine the optimum spacing of each pot.

The violet outline is on the Bottom Silkscreen layer. It will be printed on the bottom of the board and gives you a visual indication of the direction the pot is mounted. The VR## text box will be populated when the program annotates the parts on the schematic. The small circle with the + symbol to the left of the hole indicates the location of the anti-rotation key on the pot. This key is a small extrusion of metal on one side of the pot that drops into a hole on the panel to keep the pot from rotating. I added this symbol to the silkscreen so I know where to drill the ⅛" hole partially through the board for the key to drop in to.

NOTE: Anti-rotation keys are also used on the toggle switch washers to keep the switch from rotating. Don't drill the key holes all the way through the PCB or they will be visible from the top. You may need to trim some of the metal off the key since it's designed to go all the way through the panel.

The three green squares at the bottom are assigned numbers that correlate to the pins on the schematic drawing of the device. They are on the bottom copper layer. The ordering of these numbers is critical as they also indicate the pot's turning direction. As indicated on the datasheet the resistance between pins 1 and 2 is 0 Ω when the pot is fully counter-clockwise. These pads will be connected to other devices or the pin header via traces on the bottom layer. I chose to make the pads 2 mm by 2 mm square to allow ample room to solder the jumper wires from the lugs on the pots. I located them sufficiently away from the pot's body to allow room to work with the jumpers.

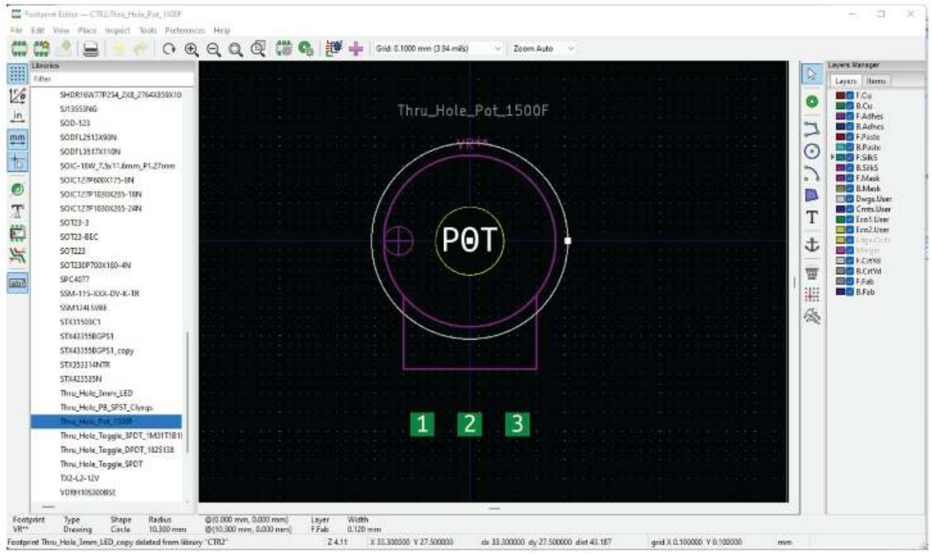


Figure 5 — Screen shot of Kicad Footprint Editor.

The other custom footprints follow the same basic guidelines. My footprints are available on the **www.arrl.org/QEXfiles** web site. You can use them as a springboard for designing your own footprints.

Once you have annotated the schematic and assigned footprints to every device it's time to design the board.

Designing the Board

To design your board, start Pcbnew, the board layout editor that comes with Kicad. Do this by clicking the green circuit board icon in the schematic capture program.

The first order of business is defining the board edges. This is done on the Edge.Cuts layer. Before you start drawing the board, choose whether you want to use millimeter or inch units and set the Grid Size to 0.1 mm or 3.94 mils. I prefer to work in millimeters but switch to inches if a datasheet doesn't provide millimeter dimensions.

Next, locate a reference point on your drawing and zero the coordinate system to it. I like to start in the top-left corner of the board. Move the cursor to where you want to locate this corner then press the Space key. This will set the Relative Coordinates dx and dy to 0.0 and 0.0. Start your board from this location. Select the Add graphic line tool and draw a square to match the board size specified in the datasheet for your enclosure. Watch the dx and dy coordinates as you draw. This will help you make these edges exactly the right size. When you approach the starting location, a circle will appear around the line intersection when both ends match up. Click the mouse when the circle is visible to complete the board edge.

Next, locate the center of the board and place an alignment target there. Zero the coordinates to this location. This gives you a common reference for everything else on the board. Be forewarned that the board house may interpret this mark as a hole and drill a hole at this location so don't route any traces through this mark. Yeah, guess how I learned this lesson.

Once you've established the center of the board it's time to import footprints and the net list from the schematic capture program. This is done by clicking the Update PCB from the schematic button on the top button bar.

Once this has been done, all of the components will appear with white air wires connecting them. This is called the rats nest for good reason. Your next job is to sort this mess out! Initially all the components will be red, indicating they are assigned to the top copper layer of the board. To move them to the bottom layer, select each component and press the F key on the keyboard to flip the component. It will turn green to indicate it's assigned to the bottom copper layer. You can highlight and flip groups of components.

Mounting holes are treated just like any other component. Add them to your schematic then assign a footprint for them to define their size. In the PCB layout program, locate mounting holes first and position them exactly where the datasheet shows them using the dx and dy coordinates. Move a footprint by first clicking on it with the left mouse button to select it and then press the M (move) key on the keyboard. The component 'sticks' to the mouse cursor until you click the left mouse button again to release it.

Next, drag each major component to the location you determined in your initial scale drawing. Press the R key to rotate it to the left in 90° steps while you have it selected. Click the left mouse button to release it. Use the dx and dy coordinates to locate these components exactly where your scale drawing shows them. The quality of your finished project depends on these components being in the right place.

Take special care when designing a two board system so that components on one board don't interfere with the components on the other board. For example, on v1.0 of my controller boards, the pin header on the front panel was close to the top of the panel. I needed to use stacking 3.5 mm plug assemblies on the bottom board due to the number I/O connections I have. These stacked assemblies were tall enough that they interfered with the DuPont connectors plugged into the front panel's pin header. v1.1 of this board moved the pin header down 10 mm to eliminate this problem.

Pcbnew has a built-in 3D renderer that really helps visualize what your board will look like when it's finished. This is very helpful for finding many clearance problems.

I found that using two single-row SMT 0.1" headers worked well for my front panel board as the headers were located toward the top of the board and all the traces ran down the board from them. You can use a dual-row header if you can center it in the middle of the board and run traces both ways. Again, keep in mind that you don't want any traces crossing over other traces, and you need the ground plane to be unbroken so that every component referenced to ground has a path back to ground.

If you absolutely can't get away from having traces crossing over each other, add a 0 Ω SMT resistor to one of the lines and place this resistor on the board where it will bridge over the other trace. If you use wider traces (0.4 mm or more) you may need to use a larger SMT resistor to bridge the trace. I typically use 0603 sized SMT resistors and capacitors but go to 0805 size when I need to bridge a 0.4 mm trace.

Sometimes there is just no way around islanding a ground return as shown on **Figure 7**. You could use a couple of vias to connect the island to the front copper but this spoils the look of the panel. To get around this, add one or more 0 Ω resistors to your schematic and connect them all to ground as shown in **Figure 6**. These resistors allow you to build a bridge to your island. You can use vias in places where a control knob will hide them.

When you layout the PCB, position these resistors to bridge across the trace(s) that are blocking the ground path. R14 and R15 are circled in **Figure 7**. Your PCB layout software will probably complain about unconnected items because it doesn't understand the purpose of the resistors but the 0 Ω resistors will do their job and the LED will light. An alternate method would be to create a custom footprint for a jumper and just solder an insulated wire to each pad to form the bridge connection.

Producing the Board

Once you have the board designed, run the design rules check (DRC) to verify everything is connected and there are no clearance violations. Fix anything on the report. Next, go over the design several times manually checking the schematic and board to make sure everything is in order. It's easy to miss connecting a wire in the schematic capture program and the DRC in the PCB layout program won't catch it. Yes, another lesson learned!

Once you're confident you have it right, create the Gerber files for your board by clicking the Plotter icon. These files describe the physical layout of each layer of the board and tell the board house how to create your board. Again, there are several videos on the Internet describing this process and Kicad makes it nearly foolproof. You should have 12 Gerber files when you're done — don't forget to generate the drill files too. Zip these into a single file and upload them to the board house of your choosing. There are many Chinese board houses that provide good quality, low quantity hobby boards at very reasonable prices. I had **JLCPCB.com**

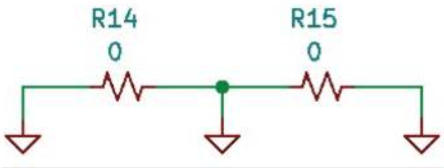


Figure 6 — Using 0 Ω resistors to connect isolated ground pads on a PCB.

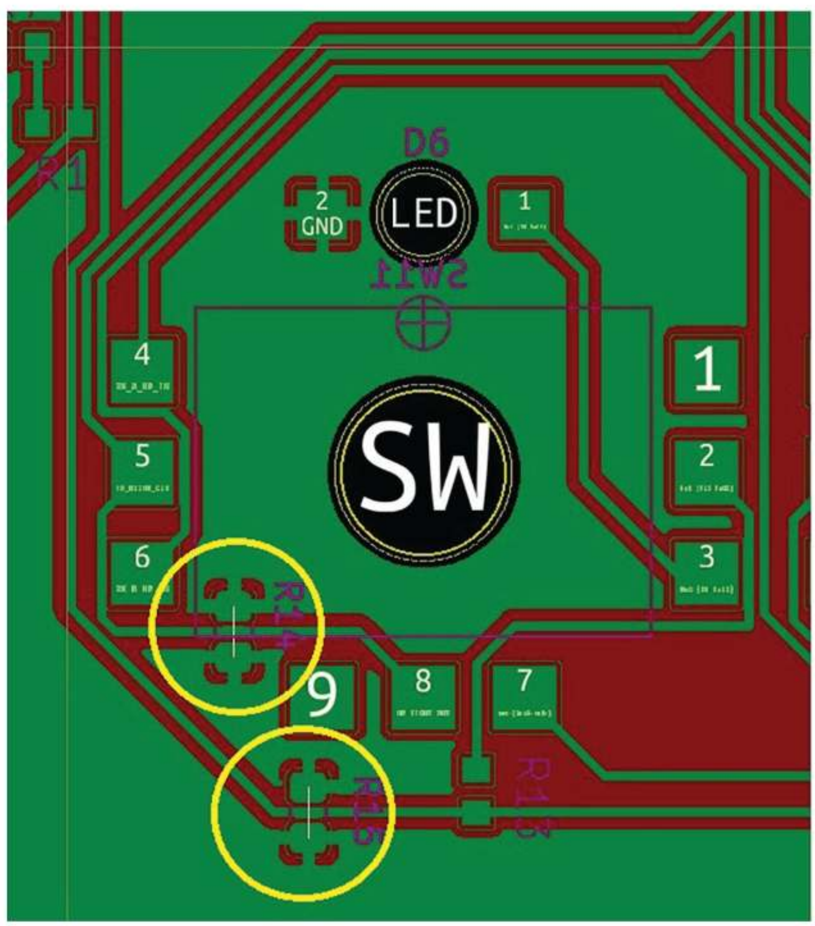


Figure 7 — Close up of two resistors connecting an isolated ground pad on a PCB.

create five of my front panel boards for $10 plus shipping. Check around with different board houses because some of them increase the price considerably if your board size exceeds 100 mm by 100 mm.

Wrapping it up

If you're lucky your new boards should show up in about two weeks and you'll be able to build a project you'll be pleased to display in your shack. **Figure 8** shows the completed front panel connected to the main controller board using 24 DuPont female/female jumpers.

Conclusion

I hope this article has given you some ideas on how you can create inexpensive, custom, professional looking front panels for your projects. If you haven't taken the plunge into circuit and board design perhaps this will inspire you to learn a new trick. Contrary to popular belief, old dogs can learn new tricks. After all, isn't that why you got interested in amateur radio in the first place? What other hobby offers lifelong learning opportunities like this one? I've been a ham for over 50 years and have never run out of things that spark my interest.

Lynn Hansen earned his Novice license WN7QYG in 1971, upgraded to General, WA7QYG, in 1972, and to Amateur Extra class, KU7Q, in 1981. Amateur radio provided a pathway into a career in electric utility communications where he spent almost 40 years, starting as a communications technician and eventually retiring as one of three operations managers over a seven state communication network. These days you can find him loving life in Lava Hot Springs, Idaho, and still learning and sharing ideas.

Figure 8 — Completed project showing the wiring interconnections between the front panel board and the main controller board.

[1] For more information on the SO2R controller and my other projects visit **https://ctr2.lynovation.com**.

[2] For more information on Kicad and to download the program, visit **https://www.kicad.org**.

[3] A via is a plated through hole in a PCB that connects the copper on the top to the copper on the bottom of the board. They are commonly used to pass a signal from one side of the board to the other or to connect ground planes together.

[4] The PacTec KEU-7 enclosure is available from Mouser and other suppliers.

[5] The PacTec KEU-7 datasheet can be found at **https://www.mouser.com/datasheet/2/314/drw_KEU-7-706759.pdf**.

[6] The data sheet for the pots I used can be found at **https://www.mouser.com/ProductDetail/313-1500F-10K**.

Peter DeNeef, AE7PD
2302 Quail Point Circle, Medford, OR 97504; **hamradioandvision@gmail.com**

Diffraction of VHF and UHF Waves By a Window

A computer model is used to estimate the diffracted fields inside a building when a transmitting antenna is located outside near a window.

In "Effects of Common Building Materials on Radio Wave Propagation," by Meredith Hillier, KG7EUM, the attenuation of a VHF signal by a 2.4 m (8 ft) square metal panel is surprisingly low because of diffraction by the edges of the panel **[1]**. VHF and UHF fields can also be diffracted by a window aperture. I used a computer model to estimate the power density of diffraction fields inside a building when a transmitting antenna is located near a window.

The simulations apply to a window aperture in metal siding, aluminum foil, or other metalized material. A non-conducting obstruction such as brick causes diffraction without the reactive near fields that are present in these simulations. Plywood, chipboard, and drywall do not absorb enough VHF or UHF radiation to cause significant diffraction **[2]**.

This analysis assumes that the window is not covered by a metal screen or a reflective glass coating.

2 m Band Diffraction Simulations

HOBBIES software for computational electromagnetics is designed to model wires and surfaces **[3]**, **[4]**, **[5]**. *HOBBIES* was used previously to calculate VHF fields inside a car with a monopole antenna on the roof. Fields enter through window apertures and excite electromagnetic resonances **[4]**, **[5]**.

Figure 1 is a diagram of a vertical half-wave dipole located 2 m (6.6 ft) from a 2.4 m (8 ft) wide, perfectly conducting panel. In this cross-section view the center of the antenna and the center of a 1.0 m (3.3 ft) square aperture are 4.2 m (14 ft) above average ground ($\varepsilon = 13$, $\sigma = 0.005$ S/m).

HOBBIES computes power density using the magnitude of the Poynting vector, which is accurate in near and far field regions. **Figure 2** shows contours of local power density *S*, W/m^2, in the *yz*-plane, perpendicular to the panel. The average transmitted power is $P = 50$ W at 146 MHz. The white area shows where *S* exceeds 2.0 W/m^2. The spatial peak of *S* is 5.0 W/m^2. The power density decreases rapidly with increasing distance from the panel – a characteristic of diffraction when the wavelength is larger than the size of the aperture. Fields from diffraction by the top and sides of the panel are negligible compared with diffraction by the aperture.

I estimate whole-body-average power

Figure 1 — Cross-section diagram of a vertical 146 MHz dipole and second-floor window aperture. The vertical 1.8 m line is an example of the locations of power density samples.

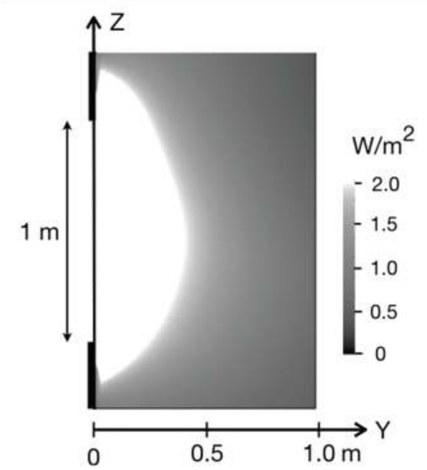


Figure 2 — Power density contours, *S*, W/m^2, for a 1.0 m square aperture when $P = 50$ W at 146 MHz. White area: $S > 2.0$ W/m^2.

density exposure, S_{AV}, W/m^2, for a standing adult by averaging samples of power density on a vertical 1.8 m line, starting at the second-floor level. **Figure 1** shows an example. The minimum separation distance *D* is measured horizontally along the center axis (parallel to the *y*-axis) from the aperture to the location where S_{AV} = 2.0 W/m^2. **Figure 3** shows *D*, m, vs. average power, *P*, W, for a 1.0 m square aperture and a 1.5 m square aperture. The dashed line shows *D* for the field of the dipole at $y = -2$ m with no obstruction. These results apply to distances that are 20 cm or more from the metal panel: At distances less than 20 cm the FCC requires compliance evaluations based on specific absorption rate (SAR). The figure shows that the size of an exclusion zone is less than 1.2 m for $P < 150$ W.

70 cm Band Diffraction Simulations

Figure 4 shows contours of the local power density for diffraction of signals in the 70 cm band by a 1.0 m square aperture. The average power $P = 100$ W at 446 MHz. The white area shows where *S* exceeds 3.0 W/m^2 At this frequency the peak power density is centered 0.45 m from the panel, independent of *P*. The power threshold where the minimum separation distance *D* (based on S_{AV}) is greater than 20 cm is 140 W.

HOBBIES Simulations

Metal surfaces in *HOBBIES* models comprise many small patches. The edges of the patches form a mesh that *HOBBIES* generates automatically. If a mesh size is small enough, computed field levels do not change significantly when the mesh size is decreased in a convergence test.

I used *NEC* v5.0 for validation tests of *HOBBIES* **[6]**. For a 1.0 m square aperture in free space the *E* and *H* diffraction fields are within 3% to 6% of the *HOBBIES* simulations. Some difference is expected because *NEC* v5.0 uses linear basis functions to model patch currents, while *HOBBIES* uses higher order polynomials for increased accuracy.

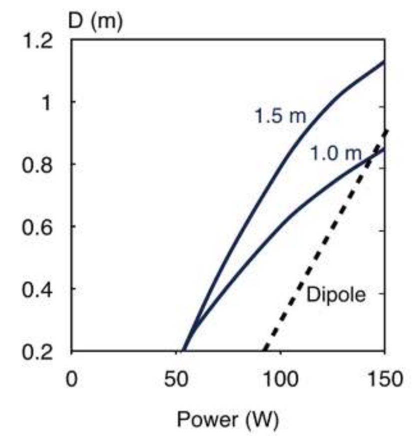


Figure 3 — Solid lines: Minimum separation distance *D*, m, vs. average power *P*, W, for a 1.5 m aperture and 1.0 m aperture at 146 MHz. Dashed line: *D* for the dipole field with no obstruction.

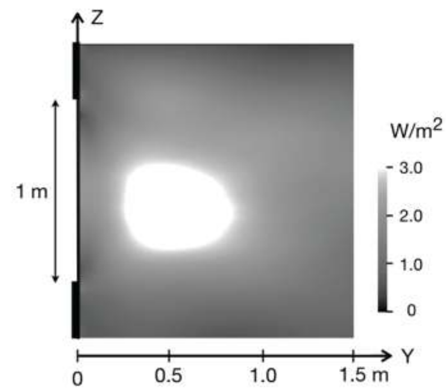


Figure 4 — Power density contours, *S*, W/m^2, for a 1.0m aperture when *P* = 100 W at 446 MHz. White area: *S* > 3.0 W/m^2.

Conclusions

In microwave and optical applications the aperture size is often 10 or more wavelengths, and near fields extend for a distance many times the aperture size **[7]**. In the 2 m and 70 cm band examples the apertures range from 0.5 – 1.5 wavelength, and near fields are more localized.

When the power is below 50 W for the 2 m band and 140 W for the 70 cm band, the minimum separation distance for these examples is 20 cm. Below 20 cm the FCC requires SAR analysis. Above the 50 W and 140 W power thresholds, the minimum distance depends on the size of the window, and it increases with power. **Figure 3** shows that the size of an exclusion zone in the 2 m band examples is $D < 1.2$ m for $P < 150$ W.

Peter DeNeef, AE7PD, received his first license as KF7FPX in 2009. He has written about RF exposure safety for QEX, as well as articles about international RF safety guidelines of the International Commission on Non-Ionizing Radiation Protection (ICNIRP). More of his articles can be found on his popular web site for vision-impaired hams, ***www.HamRadioAndVision.com***.

References

[1] M.W. Hillier, KG7EUM, "Effects of Common Building Materials on Radio Wave Propagation," *QST* Jan. 2022, pp. 35-38.

[2] Building Materials and Propagation, Ofcom report 2604/BMEM/R/3/2.0, 2014, p 40; **https://www.ofcom.org.uk/__data/assets/pdf_file/0016/84022/building_materials_and_propagation.pdf**.

[3] Y. Zhang, et al., *Higher Order Basis Based Integral Equation Solver* [HOBBIES], Wiley, 2012.

[4] S. Stearns, K6OIK, "HOBBIES Software for Computational Electromagnetics," *QEX* Nov./Dec. 2020, pp. 17-21.

[5] S. Stearns, K6OIK, "Antenna Modeling for Radio Amateurs, Revised and Expanded," presented at ARRL Pacificon Antenna Seminar, San Ramon, CA, Oct. 19-21, 2018. Slides archived at **https://www.fars.k6ya.org/docs/k6oik**.

[6] NEC v5.0 Numerical Electromagnetic Code requires a license agreement from the Lawrence Livermore Laboratory. **https://ipo.llnl.gov/technologies/it-and-communications/nec-v50-numerical-electromagnetic-code**.

[7] S. Laybros, P.F. Combes, and H. J. Mametsa, "The 'Very-Near Field' Region of Equiphase Radiating Apertures," *IEEE Antennas and Propagation Magazine* 47 (4), Aug 2005, pp. 50-66.

Steve Geers, KA8BUW
3721 Dust Commander Dr., Hamilton, OH 45011; **steveg@techdevllc.com**

DSP CW Filter

This CW audio filter demonstrates the digital signal processing capability of a microcontroller.

A Dayton Hamfest forum speaker asked, "Can you implement software-defined radio (SDR) on a microcontroller?" A microcontroller is inexpensive. The hardware design can be smaller, simpler, and lower power than a PC or Raspberry PI solution. Most microprocessor vendors supply a free integrated development environment (IDE) that includes a graphical hardware configurator and compiler tool-chain. The hardware configurator generates a code shell that takes most of the effort out of configuring the microprocessor hardware peripherals.

The CW audio filter presented in this article demonstrates the digital signal processing, DSP, capability of a microcontroller. The filter is placed between the speaker output of my Drake 2B receiver and a computer speaker. While the *Octave* computational math language files used to analyze the filtering are available on the website **www. arrl. org/QEXfiles**, the filter implementation details are presented without delving deeply into the mathematics of DSP. The microprocessor software project is also available on the website.

The project is implemented using the analog to digital converter (ADC), DSP, and digital to analog converter (DAC) functions on a low-cost microcontroller. These are combined with external amplifiers. A breadboard containing the input and output amplifiers and an STMicroelectronics NUCLEO-F334R8 microcontroller development board comprise the hardware.

The STM32F334R8 microcontroller, on the development board, is based on an ARM Cortex-M4 core that has dedicated DSP operations including a multiply and accumulate instruction. Its maximum clock speed is 72 MHz. The microcontroller has a 12-bit 5 MHz ADC and a 1 MHz DAC at a single unit cost of $7.79. The microprocessor development board contains a debugger, USB serial port, and breakout pins all for $10.

The amplifier board schematic, shown in **Figure 1**, contains two op-amp circuits. An input amplifier provides the gain from the speaker level to the ADC input range while the output circuit drives the external computer speaker. There is some anti-aliasing filtering, but that could be increased. LM8261 op-amps were selected because they were in my parts stash. Almost any op-amp, such as an LM358, could be used provided it can operate on 5 V. Duplicate contacts for test instrument connections were added to the input and output connectors. The amplifier board 5 V power is supplied by the development board via pin 4 of J2. Connections between the amplifier and microcontroller board are shown in **Table 1**. The completed hardware is shown in **Figure 2**.

The sample rate is a trade-off between maximizing the processing time between samples and providing sufficient bandwidth for audio signals. 12 kHz, the chosen sample rate, is adequate for voice filtering on future projects.

AC signals have positive and negative values. When the average voltage of a complete cycle is not zero, the difference between the average reading and zero is the dc offset. The microcontroller analog to digital converter, ADC, readings range from 0 to 4069, representing 0 to 5 V. In order to read audio signals with the ADC, the input is ac-coupled to remove any dc offset and then summed with a 2.5 V offset to place the signal into the mid-range of the ADC. Many DSP algorithms have trouble when the signal contains a dc component. A software high-pass filter is used to remove the dc. An infinite impulse response (IIR) design is implemented. The filter calculation is **[1]**:

$$w(t) = x(t) + \alpha w(t-1)$$

$$w(t) = y(t) + w(t-1)$$

where:

$y(t)$ is the output sample

$x(t)$ is the input sample

α is a feedback constant to set the high-pass corner frequency

t is the sample time.

The high-pass response has a corner frequency around 300 Hz, shown in **Figure 3**, calculated using the *Octave* script *response.m* in **QEXfiles**.

The band-pass filtering is based on a 65 tap finite impulse response (FIR) filter design. A

Table 1 - Wiring table.

Amp Pin	*J2 Signal name*	*NUCLEO Board Pin*	*NUCLEO board Micro IO Pin*
J2-1	DAC out	NC	
J2-2	DAC out	CN7-32	PA4
J2-3	GND	CN7-20	GND
J2-4	5V	CN7-18	5V
J2-5	GND	CN7-22	GND
J2-6	ADC in	CN7-28	PA0
J2-7	ADC in	NC	

direct form FIR filter is easy to implement in code, is stable, and limits the effects of round-off errors from iterative calculations. **Figure 4** shows the direct form implementation of an FIR filter **[2]**. The output sample $y[n]$, is a summation of various delay samples of the input signal, $Z{-1}$, multiplied by various coefficients. The multiply and accumulate instruction of the microprocessor efficiently performs the calculation.

Previous experiments had shown that the necessary microcontroller operations for a 65 tap filter can be implemented in the interval between samples. The filtering is based on a 65 tap FIR filter design created from a digital filter design website. **Figure 5** shows a screenshot of the design **[3]**. A 100 Hz bandwidth filter centered at 750

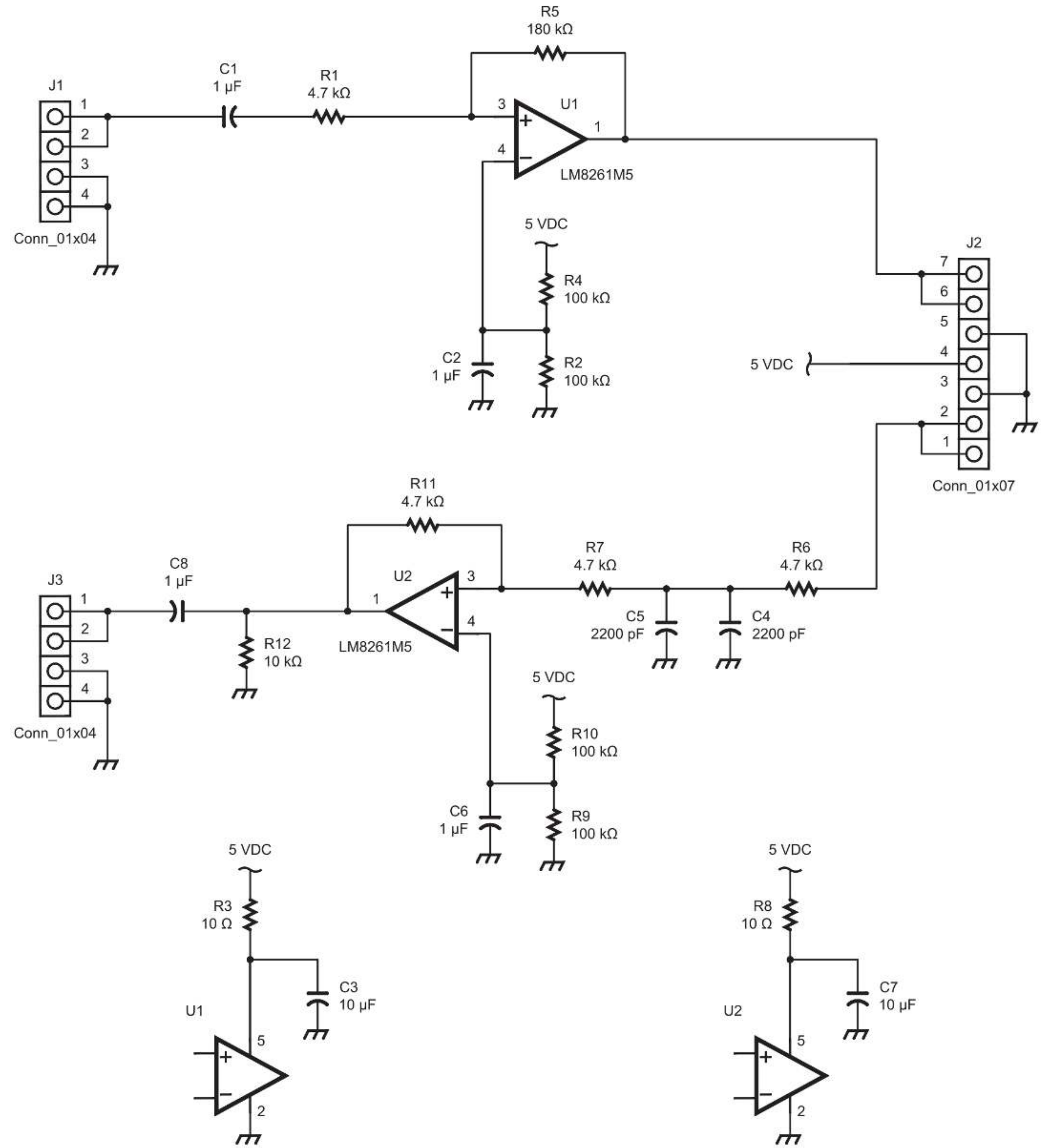


Figure 1 — Amplifier board is powered by the NUCLEO board via J2-pin4.

Hz was specified. The web application calculated the coefficients for the software implementation. **Figure 6** shows the expected response. The 3 dB bandwidth is 266 Hz while the out of band rejection is expected to be greater than 45 dB. The *Octave* script *bandpass_response.m* in **QEXfiles** calculates the response to confirm the web design. The widening of the filer response is caused by the mathematical calculation of the FIR filter. Varying the bandwidth and rejection on the website designer alters the trade off between the bandwidth and the height of the ripples in the rejection regions. Exploring this trade off is a future project.

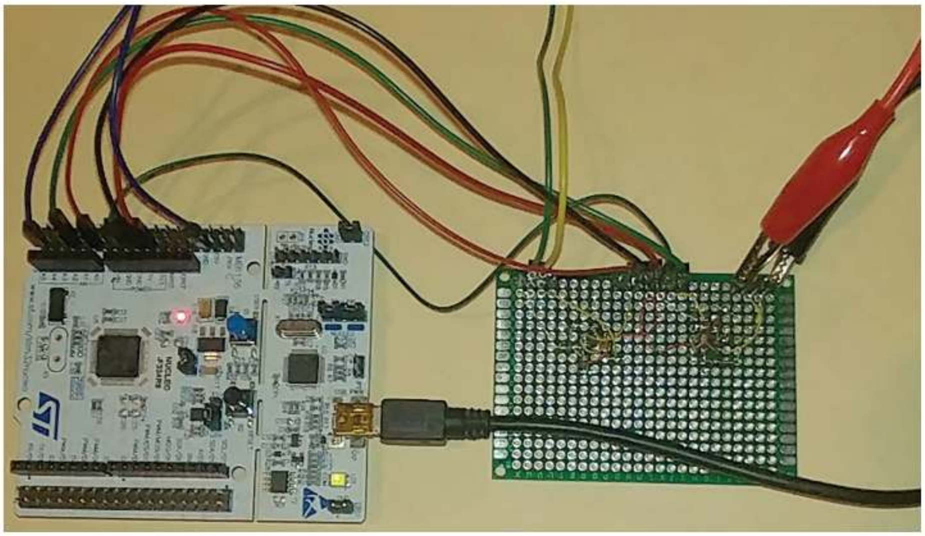


Figure 2 — Completed hardware shows the amplifier board on the right and NUCLEO board on the left with its micro connector on the lower right.

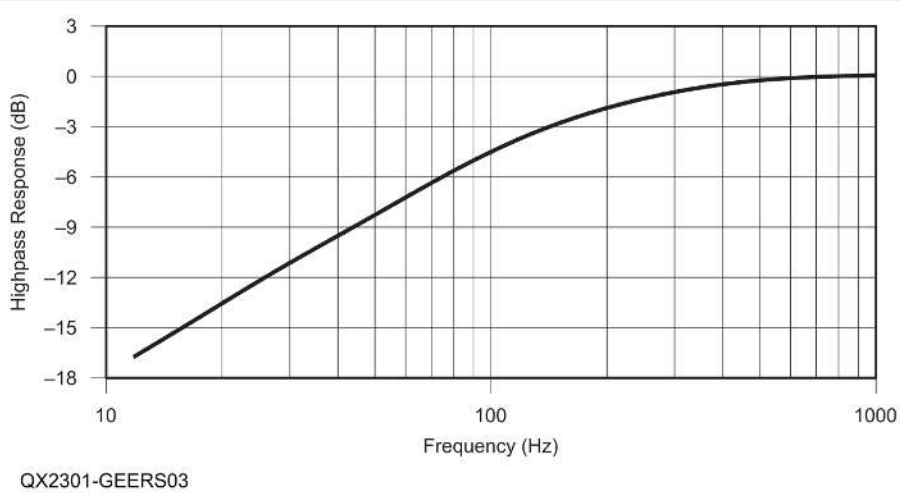


Figure 3 — Calculated high-pass filter response.

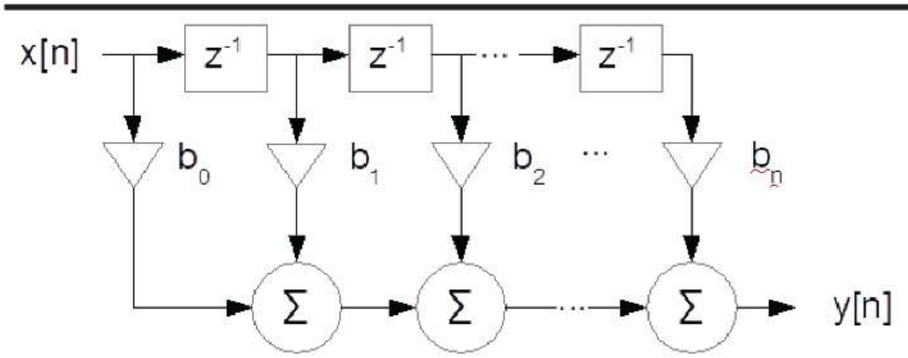


Figure 4 — The direct-form FIR filter implementation.

The M4-Cortex DSP instructions are integer based. The DSP filter coefficients from the website design were real decimal values, so they were converted to 16-bit precision in 32-bit variables. The ADC conversion is also promoted to 32-bit precision. Using 32-bit math allows sufficient headroom for the math calculations.

The complete sequence of filtering operations is shown in **Figure 7**. It begins with analog to digital conversion, then, removal of dc, next the band-pass FIR filter calculation, applying the output dc offset, and finally, digital to analog conversion.

Precise timing of the input and output samples is critical to signal integrity and is accomplished by using hardware elements of the microcontroller to control the timing. DSP is challenging because the calculations for a given sample must be performed in the interval between samples. Code must be written for fast execution. Straight line coding with no loops was written, in order to eliminate any extra operations that might be inserted for looping or array lookups.

TIM3 is a hardware timer of the microcontroller. It sets the sample rate by generating 12 kHz intervals that trigger the ADC and DAC conversions. The timer also generates an interrupt that executes the DSP code in the interrupt routine. GPIO pin C0 is set high at the beginning of the timer interrupt routine running DSP code and set low at the end of the DSP code. This allows for measuring the DSP time on an oscilloscope.

The project shell code with the peripheral hardware configuration was generated in the STM32CubeIDE using hardware settings that are configured in the IDE GUI environment. User code is then added to various defined locations in the code shell.

One of the nice features of the STM Cube-generated code is that it has clearly delineated sections to add user code to customize the operation. This allows for bypassing code that is not needed in the application and putting needed operations in critical locations. The user-added code is kept when STM Cube regenerates the application after a hardware configuration change. The complete software project is available on the **QEXfiles** website.

To speed up the operation, lower-level API functions for the ADC, DAC, and GPIO interface were used to speed up the operation of the DSP calculation. This involved stepping through the code to figure out the minimum code required for an operation. In some cases, various layers of code were combined to eliminate function calls and returns to save clock cycles. The timer 3 interrupt handler, TIM3_IRQHandler(), was generated by the STM Cube environment. It was modified to not call the remaining timer interrupt functions in HAL_TIM_IRQHandler as they are not needed. This drastically speeds up the interrupt routine. The remaining timer interrupt code is not needed by this application. Lower level GPIO routines were directly accessed to speed up the GPIO operation of the timing measurements detailed later.

Figure 8 shows the comparison between the calculated and measured frequency responses. A function generator and a PC based audio spectrum analyzer were used to measure the filter frequency response. The measured pass band is very similar to the simulated bandwidth while the out of band response is close. Notably, the deep nulls are not seen in the measured response, as the out of band measurement points are more of an average of the peaks and nulls. This is likely due to the sweeping nature of the signal

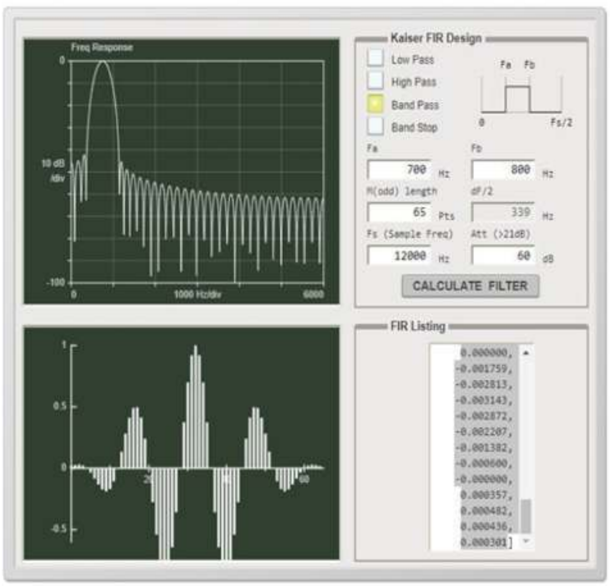


Figure 5 — FIR filter design.

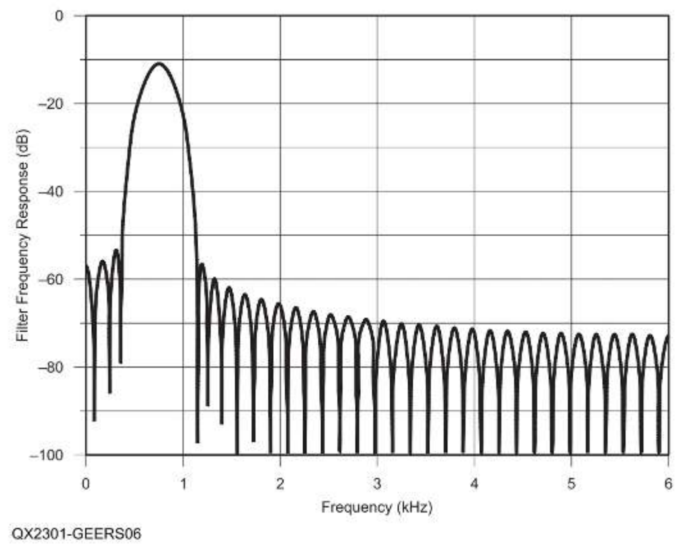


Figure 6 — Calculated band-pass response.

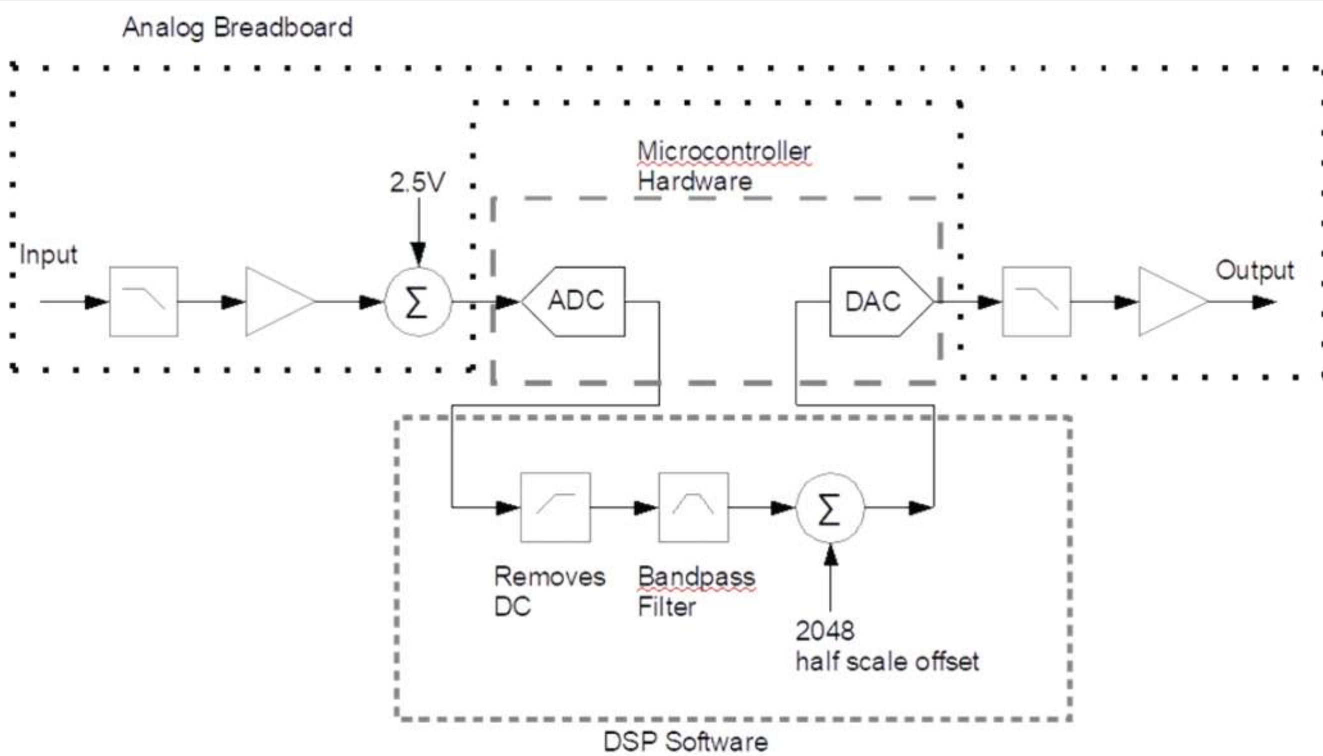


Figure 7 — Band-pass filtering system operations.

source and windowing of the spectrum analyzer software.

GPIO pin C0 is used for timing the DSP code in the interrupt function. A 700 Hz sine wave input and output on the oscilloscope are shown in **Figure 9**. Digital noise on the input is due to the low-level performance of the oscilloscope. The output signal shows the digitization of the filtered signal. By implementing more low-pass filtering, the signal could be smoothed. The square wave code trace at the bottom is the timing of a complete sample cycle. The high portion is the DSP processing time in the interrupt routine. The complete cycle is around 12 kHz as expected by the sampling rate. **Figure 10** shows the high portion of the code trace is the DSP time of 32.8 µs in the interrupt. The cycle time for 12 kHz is 83.3 µs. Less than half of the available processing time is used for DSP functions, leaving the remaining time that could be used for other DSP functions or a user interface.

The filter was tested with my Drake 2B receiver. The receiver IF bandwidth was set to 2.1 kHz. The filtering was very noticeable and removed nearby signals and a lot of broadband noise while the audio sounded very natural. Over-driving the ADC does cause a lot of distortion and should be avoided.

A CW filter has been implemented on a low-cost microcontroller with a minimal amount of extra circuitry, using a free development environment provided by the microcontroller vendor, and filtering designed from online sources. The measured performance matched the predicted performance and performed as expected with my Drake 2B receiver.

Steve Geers, KA8BUW, holds an Amateur Extra class license and has been licensed since 1978. He received a Bachelor's of Electrical Engineering from the University of Dayton and is a licensed professional engineer. He spends his time lurking in the workshop experimenting with low cost radio projects. Steve has worked in the commercial and defense industry as an RF engineer and embedded hardware and software developer.

Notes:

[1] **sam-koblenski.blogspot.com/2015/11/everyday-dsp-for-programmers-dc-and.html**.

[2] **https://en.wikipedia.org/wiki/Finite_impulse_response**.

[3] **https://www.arc.id.au/FilterDesign.html**.

Figure 8 — Calculated and measured frequency response.

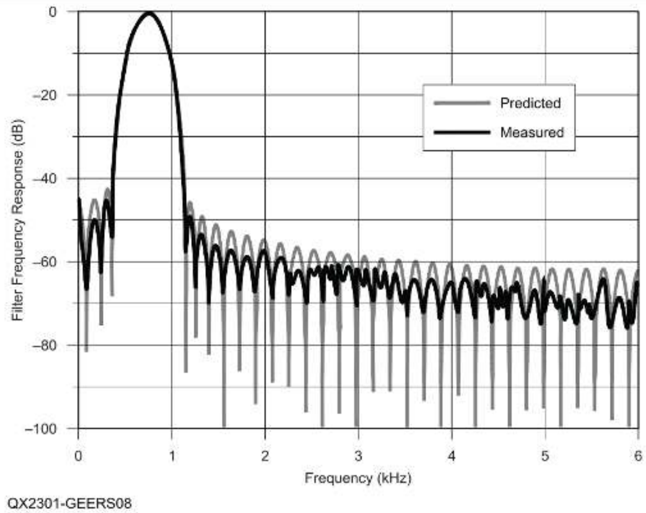


Figure 9 — A 700 Hz sine wave IN and OUT on the oscilloscope.

Figure 10 — DSP processing timing signals.

Brian Robert Callahan, AD2BA

110 8th St., ITWS, Lally Bldg. 2nd Fl., Troy, NY 12180; **callab5@rpi.edu**

Zhemin "Hisen" Zhang, KD2TAI

110 8th St., ITWS, Lally Bldg. 2nd Fl., Troy, NY 12180; **hisenzhang01@gmail.com**

An AI-powered Transcription Bot for FM Transmissions

An Artificial Intelligence and Machine Learning bot transcribes heard audio into text.

As amateur radio transmissions are sent unobscured, it is possible to perform any number of transformations on these transmissions. The burgeoning fields of Artificial Intelligence (AI) and Machine Learning (ML) have the capacity to transform what can be done with all data in amateur radio and far beyond. We believe it is prudent for the amateur radio community to embrace the potential of AI/ML while at the same time having frank conversations about its proper role in amateur radio and establishing best practices for the use of AI/ML in amateur radio.

In this article, we present a first experiment combining AI/ML and amateur radio. We will demonstrate an AI-powered bot that can listen to FM transmissions, transcribe the audio heard into text, and then post that text to a social media account, all without direct human intervention. We will begin with a short explanation of AI/ML to introduce the concepts to the amateur radio community at large. We will then document an experiment creating an AI-powered bot for FM transmissions. We will then discuss issues of accuracy. We will conclude with the opening to a much larger discussion tasking the amateur radio community to consider the broader applicability and role of AI/ML in amateur radio, ending with a call to organize for serious discussion around these AI/ML technologies.

It is our belief that AI/ML represents a new, exciting avenue of exploration for radio research and exploration, advancing the state-of-the-art in radio, a core charge for amateur radio. We believe in the potential contained within the intersection of AI/ML and amateur radio to make amateur radio exciting for the next generation of amateur operators, expand access to amateur radio, and create new opportunities for amateur operators around the world. We hope that this article inspires others to experiment with AI/ML applications in amateur radio.

What AI/ML is

Here, we provide a high-level explanation of AI/ML relevant to introducing the concepts to the amateur radio community at large and for our purposes relevant for this article's experiment.

Artificial Intelligence is well-described as "the science and engineering of making intelligent machines, especially intelligent computer programs" **[1]**. AI itself traces its roots to the early days of the discipline of computer science, with Alan Turing devising his "Imitation Game," what we now call the Turing Test, in order to answer the question of whether or not machines can think **[2]**. Much more broadly speaking, we can think of AI as being divided into a few main camps: the human approach, where the goal is to produce machines that think like humans; and the ideal approach, which is interested in producing machines that act rationally **[3]**. In today's world, AI powers much of our daily interactions, even if it is at times invisible. Apple's Siri **[4]**, Microsoft's Cortana, Google Home, and Amazon's Alexa **[5]** are examples of AI systems that you may routinely interact with. AI powers much of our decision making, from mundane questions about what to watch on Netflix to judicial scenarios **[6]**.

Machine Learning is a branch of AI that devises methods for machines to imitate the ways in which humans learn, allowing machines to self-improve their accuracy at their assigned tasks over time. Originally coined in 1959 as a term to describe a machine designed to play checkers **[7]**, machine learning today is perhaps most visible in its applications in self-driving cars **[8]**. In general, ML requires three components: a decision process that takes in data and "guesses" what the output should be; an error function that measures how good the "guess" was against known examples; and an optimization process that updates how the decision process makes its "guesses" so that future errors will not be as large **[9]**.

For our experiment in this article, we will be using an open source off-the-shelf AI/ML software package known as VOSK **[10]**, a speech recognition toolkit. While we will not be diving into the technical details surrounding the AI/ML that make VOSK work, we can be confident that the system does work, as our experiment will demonstrate.

An experiment: an AI-powered transcription bot for FM transmissions

To get set up with the AI bot required a number of steps to correctly set up all the hardware and software. This required identifying appropriate hardware and software, acquiring an SDR kit, testing and configuring software, and setting up a social media account to post transcribed recordings. All code and scripts that we wrote are open source and are available online at **https://github.com/w2sz/HamBot**.

We also set up a social media account on the website **https://mastodon.radio**. Mastodon is a Twitter-like social media service using 100% open source software and open source protocols for communication. It can be thought of as many small Twitter clones that can all communicate with one another. Many of these Mastodon sites are themed; **Mastodon.radio** is themed around amateur radio. With permission of the site owner, we created an account named **@FMbot@mastodon.radio**. This account hosts the FM transcriptions that the bot creates and posts. The posts can be read at **https://mastodon.radio/@FMbot**. **Figure 1** provides a high-level visualization of the process undertaken by the bot.

Here, we outline the steps we took so that others may replicate our bot:

1 – First, we had to select a system sufficient for running all the needed software. We used the setup outlined below for development and testing. While you may get adequate results with less powerful hardware, we cannot make any guarantees. Generally, a more powerful machine allows you to run more sophisticated models, which increases the recognition, and ultimately transcription, accuracy. In our tests, a laptop with an Intel i5 processor and 8GB RAM could not run the most accurate VOSK model.

In addition, software is needed for Python, VOSK, an audio driver to connect Python to audio hardware to enable recording, and a method for posting the transcribed text to a social media account.

a. Hardware specifications
   - i. Processor: AMD Ryzen 7 3800X 8-Core Processor 3.90GHz
   - ii. Installed RAM: 32 GB
   - iii. Software-Defined Radio: RSP1 kit connected via USB

b. Windows specifications
   - i. Edition: Windows 10 Home
   - ii. Version: 2004

c. Software specifications
   - i. Python 3.9.12
   - ii. VOSK 0.3.42
   - iii. PyAudio 0.2.11
   - iv. Mastodon.py 1.5.1
   - v. SDRuno 1.41.1
   - vi. VB-CABLE 1.0.3.5

2 – Install the Python environment and relevant packages. The ones listed below with specific version numbers were tested to work. Note that installing PyAudio on Windows can be tricky since a prebuilt binary file is unavailable from the default download location, so you may need to download PyAudio from **https://www.lfd.uci.edu/~gohlke/pythonlibs/** and manually install the package.

After that, you will need to make one edit to the PyAudio site-package: locate the *read()* function in *pyaudio.py* and set the default value for argument *exception_on_overflow* to *False* **[11]**.

3 – Acquire our demo script repository. Clone **https://github.com/w2sz/HamBot** to your local machine to get our *FMBot.py* script.

4 – Choose the VOSK model. You can find a list of models here: **https://alphacephei.com/vosk/models**. While VOSK supports many languages, in this experiment we focused on three English recognition models. The first model, vosk-model-en-us-0.22, was tested on the hardware setup listed above; this model required about 5 GB of RAM at runtime. The second model, vosk-model-en-us-0.22-lgraph, may be an alternative that fits most laptops. On devices such as smartphones or Raspberry Pi, the third model, vosk-model-small-en-us-0.15, would be the best choice.

To install the model, simply unzip it. We recommend keeping the decompressed model in the same directory as the *FMBot.py* script.

5 – Configure SDR software settings. We used SDRuno with RSP1 in NFM mode at 10 kHz bandwidth with noise reduction turned on. Virtual audio cable software *VB-CABLE* pipes the demodulated audio from SDRuno to VOSK at 44100 Hz sampling rate with max latency of 2048 samples.

6 – Acquire a Mastodon API token. You may skip this step if you only want the bot to work offline and not post to Mastodon. Once registered and logged in to Mastodon, go to Preferences → Development → click on "New application" → Fill in the fields and grant write access → Click "Submit." The string after "Your access token" is the access token you will need for the *FMBot.py* script. Replace the placeholder near the top of the *FMBot.py* script with your access token where it says 'YOUR_ACCESS_TOKEN'.

7 – Now you can run the bot. First, run the *FMBot.py* script with the "-l" argument to list all audio devices, select the audio device you want to use, then rerun the *FMBot.py* script specifying your audio device with the "-d *n*" argument, where *n* is the device index. By default, the network feature is disabled. To post content online, specify the API base URL to your Mastodon server with "--apibaseurl YOUR_MASTODON_URL".

Table 1 shows a computer transcript of what you would expect to see if everything is

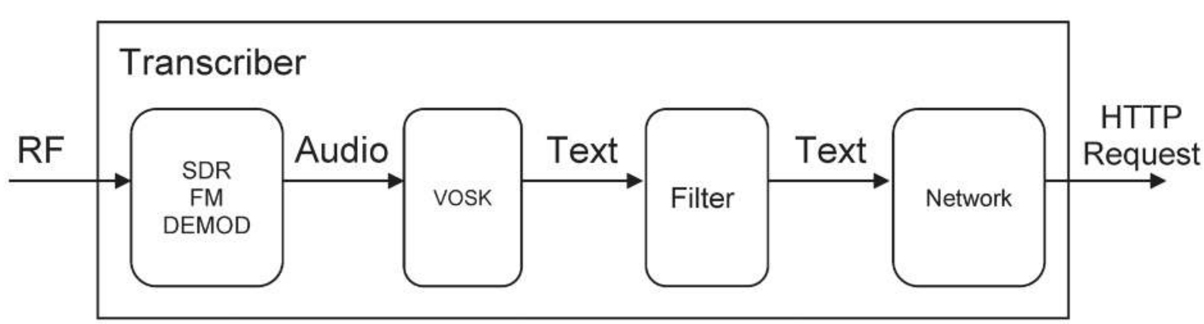


Figure 1 — System block diagram.

running correctly. This computer transcript contains a transcribed test transmission from AD2BA. Note that this test used an older script; users should replace *test_microphone.py* with *FMBot.py* in the command line invocation.

As the transcript demonstrates, the bot works. However, a quick perusal of the computer transcript identifies some clear misses and some interesting quirks in transcription. For example, in the call sign AD2BA, the bot never identifies "2" as a number but as the unrelated but homophonic word "to." Larger misses include transcribing "QEX" as "kiwi eggs."

We also discovered during our testing that VOSK may mistranscribe noise even without the presence of any audio signal. Empirically, such output produces the word "the" by itself, so we simply reject results with only one word. However, in a more sophisticated system, the filter should also examine if the content is sufficiently informative and appropriate before posting them into the public domain.

Testing the accuracy of the system

In an analog system, static noise in the background grows as the received RF signal strength decays. In addition, SSB signals are more prone to noise interference than FM. Since conditions like this were not considered during the VOSK model training, we would expect reduced transcription accuracy. It might well be germane to produce VOSK models specifically trained on amateur radio transmissions to improve accuracy.

To identify the impact of noise on our setup's speech recognition accuracy, we performed the following experiment: starting with a clear audio signal (SNR >

Table 1 – Computer Transcript.

```
$ python test_microphone.py
LOG (VoskAPI:ReadDataFiles():model.cc:213) Decoding params beam=13 max-active=7000 lattice-beam=6
LOG (VoskAPI:ReadDataFiles():model.cc:216) Silence phones 1:2:3:4:5:11:12:13:14:15
LOG (VoskAPI:RemoveOrphanNodes():nnet-nnet.cc:948) Removed 0 orphan nodes.
LOG (VoskAPI:RemoveOrphanComponents():nnet-nnet.cc:847) Removing 0 orphan components.
LOG (VoskAPI:ReadDataFiles():model.cc:248) Loading i-vector extractor from vosk-model-en-us-0.22-lgraph/ivector/final.ie
LOG (VoskAPI:ComputeDerivedVars():ivector-extractor.cc:183) Computing derived variables for iVector extractor
LOG (VoskAPI:ComputeDerivedVars():ivector-extractor.cc:204) Done.
LOG (VoskAPI:ReadDataFiles():model.cc:282) Loading HCL and G from vosk-model-en-us-0.22-lgraph/graph/HCLr.fst vosk-model-en-us-0.22-lgraph/graph/Gr.fst
LOG (VoskAPI:ReadDataFiles():model.cc:303) Loading winfo vosk-model-en-us-0.22-lgraph/graph/phones/word_boundary.int
################################################################################
Press Ctrl+C to stop the recording
################################################################################
15:38:07 the
15:38:27 the
15:38:48 the

<A few more "the" being mistranscribed>

WARNING (VoskAPI:LinearCgd():optimization.cc:549) Doing linear CGD in dimension 100, after 15 iterations the squared residual has got worse, 889.95 > 440.619. Will do an exact optimization.
15:39:30 this is alpha delta to bravo alpha conduct dna test of the fm bot with the settings on wide at them to see if it works any better than when we'd had it before on narrow fm
15:39:52 the
15:40:10 the
15:40:31 the

<A few more "the" being mistranscribed>

15:42:11 the alpha delta to bravo alpha conducting yet another test of the bought this time we are taking a fancy picture for qx since we plan to send what we've done to kiwi eggs
15:42:35 for an article and his way we have some fun and exciting pictures for everybody to see if you've been listening on the channel i guess i'm sorry but i'm with a student and we've been conducting research so once again the did alpha delta to bravo alva i'll probably come back on in the couple
15:42:43 minutes to use the more dead than but until then eighty to be a
15:43:00 the
15:43:19 the

<A few more "the" being mistranscribed>

15:45:29
WARNING (VoskAPI:LinearCgd():optimization.cc:549) Doing linear CGD in dimension 100, after 15 iterations the squared residual has got worse, 1344.06 > 383.455. Will do an exact optimization.
15:45:42 the the alpha delta to bravo alpha we've added a little more settings to the bot to see if this works any better
15:45:52 so i guess we're going to find that in the second when i walk over to the computer and the with the transcription looks like a d to be a

<End of transcript>
```

64 dB), we manually added different levels of noise to it. The SNR value measured in decibels is defined as:

$$\mathrm{SNR} = 10\log_{10}\left(\frac{P_{signal}}{P_{noise}}\right)$$

or, given that power of signal and noise are measured in decibels,

$$\mathrm{SNR} = P_{signal} - P_{noise}$$

Note we measure the SNR of demodulated audio instead of RF signal to make results comparable. This is because the performance of demodulators, either implemented in software or hardware, makes a difference. The power of the signal is the measure for the letter "G" whereas the power of noise is for the silence after it.

The resultant audio SNR ranges from 0.016 dB (noisy) to 64 dB (clear). **Figure 2** shows such waveforms for SNR = 0.016 dB, 4 dB, and 64 dB, from top to bottom. The VOSK recognizer identifies the words and outputs one word per line. Finally, we use:

$ diff –y --suppress-common-lines standard.txt result.txt | wc –l

to calculate the number of different lines, then manually apply adjustments if required.

We used the NATO phonetic alphabet from Wikipedia for the test, a recording of a female voice reading the NATO phonetic alphabet in English **[12]**. The set comprises 36 words, including letters A-Z and numerals 0-9. Note that some digits are read differently than in standard English to improve intelligibility over noisy channels, for instance, "three" as "tree" and "nine" as "niner." When calculating accuracy, we regard both results as correct. We chose this set because amateur operators exchange call signs in this format. However, to further examine the impact of SNR, we need more recordings such as amateur radio traffic from a repeater.

A higher SNR yields higher accuracy in word recognition, see **Figure 3**. The 50% cutoff for VOSK is around 2 dB. While the intelligibility of words remains largely subjective to the listener and strongly associated with one's language abilities, even a signal under 0.016 dB SNR sounds intelligible to the authors, a significant gap between human and machine.

Implications

It is widely argued that AI/ML is not value-neutral **[13, 14, 15, 16]**. Indeed, this idea is so firmly established that it even has its own term: algorithmic bias **[17]**. Even without algorithmic bias, AI/ML has the potential to cause harm to humans. Even our relatively innocuous experiment is not without the potential for harm, as an ethics question quickly arises. Do amateur operators consent to having their transmissions recorded, transcribed, and archived on the public internet for all to see? While the FCC clearly states that interception and divulgence of amateur radio transmissions is legal **[18]**, that alone may not be sufficient to not cause concern. Indeed, this concern is not theoretical: for a number of years, a local repeater frequented by the authors has had its transmissions simulcast over the internet on the website **Broadcastify [19]** without the expressed permission of any amateur operator who uses the repeater. We do not even know who set up the simulcast.

It is an inevitability that AI/ML will be used, if it is not already being used, on amateur radio transmissions and data. As amateur operators, we must not be left in the dark as to these technologies. We must have a voice as to what practices we should consider acceptable concerning the intersection of AI/ML and amateur radio. We call upon the amateur radio community to begin to debate and develop, and ultimately publish, AI/ML best practices for amateur radio immediately. We cannot afford to wait.

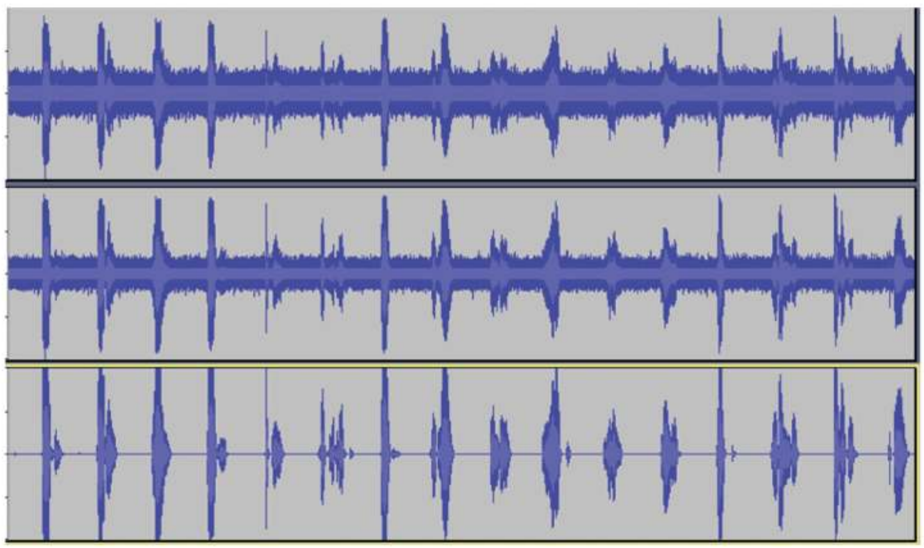


Figure 2 — Audio waveforms with different levels of added noise. SNR from top to bottom: 0.016 dB, 4 dB, and 64 dB.

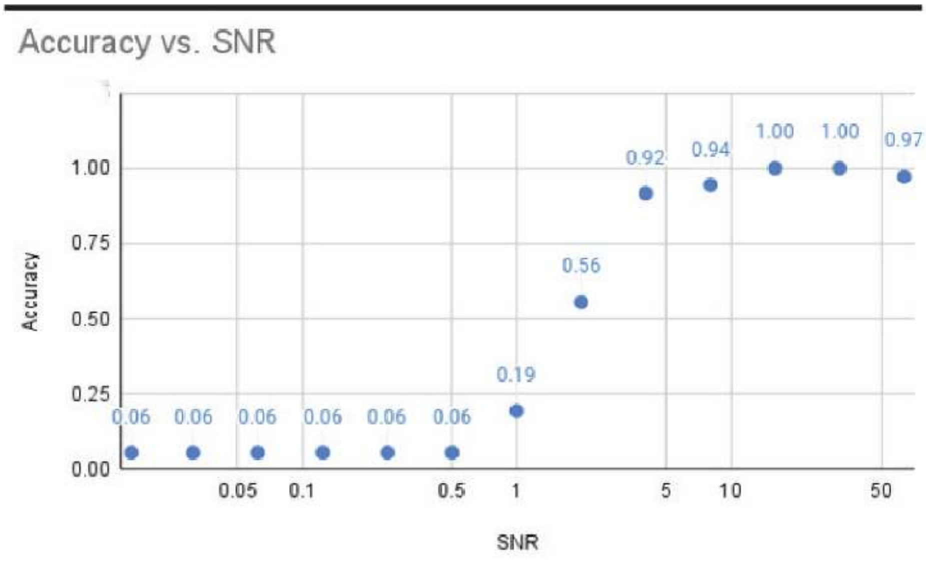


Figure 3 — Recognition accuracy vs. SNR in dB.

With that said, AI/ML has extremely positive implications as well that will enhance the amateur radio community. To explicate just one example, future articles will document how to expand the experiment in this article to be used during contesting, which we hope has the positive impact of opening up voice-based contesting to amateur operators who otherwise might not be able to compete in voice-based contesting. Another interesting use case for AI/ML within amateur radio would be to provide a method to allow clubs to meet on their local repeaters. During the pandemic, many clubs turned to Internet video streaming services for meetings. A bot like the one we describe in this article would allow clubs to meet "on the air" without needing someone to furiously transcribe notes or record the entire audio of the meeting. Finally, the bot can act as an intermediate step in developing more complicated amateur radio AI systems. We look forward to seeing what systems can be built on top of this bot.

Figure 4 — Brian Callahan, AD2BA, in the background on the RPI campus talking into an HT. You can see the transcription bot in the foreground listening to AD2BA's transmission and transcribing the audio into text.

Future Work

Future work will explore the impacts of bandwidth, accents, and other factors likely to play a role in affecting the accuracy of transcription, with possible countermeasures. Like the telephone system, the analog FM transmits only a limited portion (300 Hz - 3 kHz) of the audible spectrum to save bandwidth. Some radio hardware may apply low cutoff below the vocal frequencies to reduce noise. The impact on accuracy due to varying high and low cutoffs may be worth further exploration.

Accents may introduce a significant difference in accuracy as well, as suggested by our preliminary experiments. The aforementioned VOSK models are trained with generic US English materials but there is no detailed information about the samples.

Conclusion: The AI-powered future of amateur radio

Artificial Intelligence and Machine Learning applications represent exciting new avenues for amateur radio research and exploration, furthering the state of the art in radio. While AI/ML must be discussed within the amateur radio community in order to establish best practices for its use, we are confident that the community will rise to the challenge and establish such best practices that enhance the value of amateur radio worldwide.

In this article, we presented a first experiment to combine AI/ML and amateur radio: an AI-powered transcription bot for FM transmissions. This bot, combining open source off-the-shelf AI/ML audio transcription software with software-defined radio systems, was successfully able to listen to FM transmissions, transcribe the audio heard, and post the transcriptions to a social media account, all without human intervention.

We began by briefly explaining AI/ML for new audiences to introduce the concepts to the amateur radio community at large. We then walked through the process of setting up the hardware and software, documented a sample session, and demonstrated the end result of text posts to a social media website. We then discussed factors surrounding the accuracy of the transcriptions.

We finished by having a short discussion of the implications of AI/ML applied to amateur radio. While the implications will require serious discussion, we are hopeful that the benefits of AI/ML applied to amateur radio will serve to strengthen amateur radio, not hinder it. We envision AI/ML use in amateur radio bringing in more amateur operators and expanding access to amateur radio by amateur operators who may not be able to access all the services amateur radio has to offer.

The AI-powered future of amateur radio is bright. We look forward to learning and collaborating with other amateur operators to use AI/ML in their own amateur radio endeavors.

Acknowledgments

The authors would like to thank Christopher J., MØYNG, for hosting the **Mastodon.radio** website and allowing us to create the **@FMbot** account on the site.

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. He is the faculty advisor for W2SZ, the amateur radio club at RPI.

Zhemin "Hisen" Zhang, KD2TAI, is the president of the W2SZ college amateur radio club at RPI. He is currently pursuing a B.S. in Computer Science dual Computer and Systems Engineering, focusing on digital communication and AI technologies. He has been an Amateur Extra class operator since 2022.

Reference

[1] J. McCarthy, 2004. "What is Artificial Intelligence?" Available online: **https://www-formal.stanford.edu/jmc/whatisai.pdf"**.
[2] A. M. Turning. "Computing Machinery and Intelligence.," Mind 49: 433-460, 1950.
[3] S. Russell and P. Norvig. *Artificial Intelligence: A Modern Approach, Third Edition*, Prentice Hall: Upper Saddle River, NJ, 2010.
[4] A. Kaplan and M. Haenlein. "Siri, Siri, in my hand: Who's the fairest in the land? On the interpretations, illustrations, and implications of artificial intelligence," Business Horizons 62(1): 15-25. 2019.
[5] V. Këpuska, and G. Bohouta, "Next-generation of virtual personal assistants (Microsoft Cortana, Apple Siri, Amazon Alexa and Google Home)," 2018 IEEE 8th Annual Computing and Communication Workshop and Conference (CCWC): 99-103, 2018.
[6] J. J. Ferreira and M. de Souza Monteiro, "Evidence-based Explanation to Promote Fairness in AI systems," CHI 2020 Extended Abstracts. Apr. 25-30, 2020. Honolulu, HI. Available online: **https://arxiv.org/ftp/arxiv/papers/2003/2003.01525.pdf**.
[7] A. L. Samuel, "Some Studies in Machine Learning Using the Game of Checkers," IBM Journal of Research and Development 3(3): 210-229, 1959.
[8] J. Stilgoe, "Machine learning, social learning and the governance of self-driving cars," Social Studies of Science 48(1): 25-56, 2017.
[9] UC Berkeley School of Information. 2020. "What is Machine Learning (ML)?" Available online: **https://ischoolonline.berkeley.edu/blog/what-is-machine-learning/**.
[10] **https://alphacephei.com/vosk/**.
[11] KanzaSheikh, "Offline Speech Recognition Made Easy with Vosk," 2020, Available online: **https://medium.com/analytics-vidhya/offline-speech-recognition-made-easy-with-vosk-c61f7b720215**.
[12] **https://en.wikipedia.org/wiki/File:NATO_Phonetic_Alphabet_reading.ogg**.
[13] B. Friedman and H. Nissenbaum , "Bias in computer systems," ACM Transactions on Information Systems 14(3): 330-347, 1996.
[14] R. Kitchin, "Thinking critically about and researching algorithms," Information, Communication & Society 20(1): 14-29, 2017.
[15] N. Seaver, "Knowing algorithms," In *digitalSTS: A Field Guide for Science & Technology Studies*, Janet Vertesi & David Ribes, eds. Princeton: Princeton University Press, pp. 412-422, 2019.
[16] C. Stinson, "Algorithms are not neutral," AI Ethics, 2022, **https://doi.org/10.1007/s43681-022-00136-w**.
[17] **https://en.wikipedia.org/wiki/Algorithmic_bias**.
[18] **https://www.fcc.gov/consumers/guides/interception-and-divulgence-radio-communications**.
[19] **https://www.broadcastify.com/listen/feed/32220**.

Upcoming Conferences

TechCon 2023

Tampa, Florida
February 24 – 25, 2023

arrlwcf.org/wcf-special-events/wcftechconference

The 9th Annual ARRL West Central Florida Section Technical Conference — TechCon, will be held February 24 – 25, 2023, at the Hillsborough County Public Safety Operations Complex – Emergency Operations Center, Tampa, Florida. See website for information.

SCALE 20x

Pasadena, California
March 9 – 12, 2023

www.socallinuxexpo.org/scale/20x

The 20th Annual southern California Linux Expo, SCALE 20x, will take place on March 9 – 12 2023 at the Pasadena Convention Center in Pasadena, California. See website for information.

2023 SARA Western Conference

Bishop, California
March 17 – 18, 2023

www.radio-astronomy.org

The 2023 SARA Western Conference will be held at the Owens Valley Radio Observatory, Bishop, California, March 17-18, 2023. See website for information.

Microwave Update 2023 and 46th Eastern VHF/UHF Conference

Windsor, Connecticut
April 14 – 15, 2023

www.microwaveupdate.org

Microwave Update 2023 and the 46th Eastern VHF/UHF Conference will be held April 14 – 15, 2023, at the Hilton Garden Inn at Bradley Airport, Windsor, CT. See website for information.

Dr. Ulrich L. Rohde, N1UL, Elected Fellow of INAE

Taking note of his outstanding contributions to engineering and also his dynamic leadership in the engineering domain which have immensely contributed for the faster development of the country, the Indian National Academy of Engineering (INAE) Council has decided to elect Dr. Ulrich L. Rohde as a Fellow of the INAE in 2022. The Academy covers the entire engineering profession and aims at the promotion and advancement of the practice of engineering and technology and their applications to problems of national importance.

QEX joins in congratulating Dr. Ulrich L. Rohde, N1UL, a frequent *QEX* author, on achieving such a high honor.

Eric P. Nichols, KL7AJ
AlasKit Educational and Scientific Resources, 3763 Lyle Ave., North Pole, AK 99705; **kl7aj72@gmail.com**

Self-Paced Essays — #15 Filters

Exploring filters, the most basic of ac circuits.

This month we'll backtrack a little bit and look at the response of filters, which should give some deep insights into all things electronic. In "Almost AC," March/April 2022 *QEX* we explored a simple *RC* circuit (**Figure 1**), which is actually a low-pass filter.

The time constant of this filter is 0.1 ms. Now, instead of driving this circuit with a dc voltage pulse, let's drive it with a frequency swept ac signal, in this case, 100 to 20,000 Hz and look at the frequency domain response in **Figure 2**. The output is the voltage at the junction of R1 and C1.

As the frequency increases, the reactance of C1 decreases, and shunts more of the current to ground. R1 and C1 actually form a voltage divider, with R1 being a constant, and X_{C1} being a variable. Note the frequency at which the voltage drops to 0.707 times (there's that magic number again) the low frequency amplitude. If there were a load, this would be the half-power point. This is the cutoff frequency. Clearly, the output doesn't abruptly cut off at this point, but it's the official "starting point" of where filter action kicks in. In a perfect filter, the output would drop to zero right at the cutoff frequency, but this is an unobtainable goal. Filter design for decades has involved the attempt to approach this ideal as closely as possible. It is no mean feat to do so.

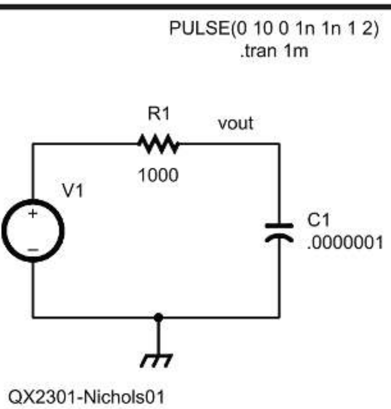


Figure 1 — A pulse driven *RC* circuit.

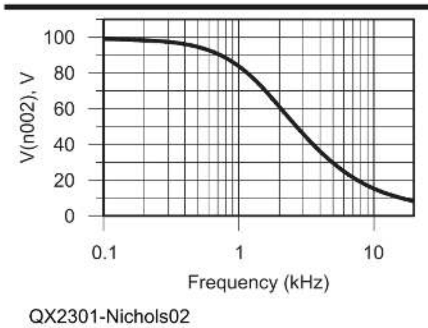


Figure 2 — Voltage at the junction of R1 and C1.

Let us step back a moment and relate our cutoff frequency to the *RC* time constant. The simple formula for this is, $f_c = 1/(2\pi RC)$ where f_c is the in Hz, *R* is in ohms and *C* is in farads.

Plugging in our numbers we see how this checks out:

$1/(1000 \times 0.0000001 \times 2\pi)$ gives us 1592 Hz. Eyeballing the graph, we see that at 1592 Hz, the voltage is about 71 V. The frequency scale is logarithmic, so eyeballing it isn't perfect, but will give us a good sanity check, anyway.

We could call this circuit unloaded, because we haven't given any consideration to what's hanging off the far end. The response of any passive filter will always be affected by the what's on the load (output) end, and so all proper passive filters are designed for a specific load impedance.

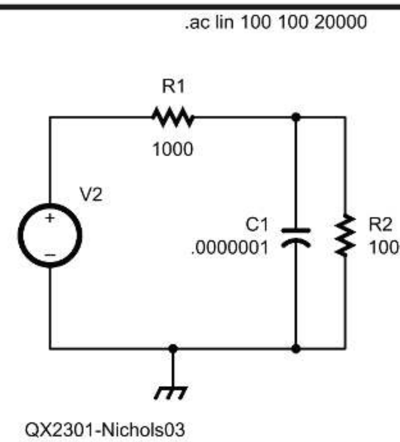


Figure 3 — An ac circuit with a load.

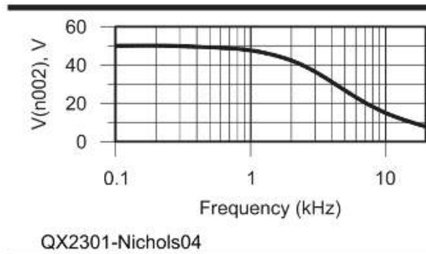


Figure 4 — Output of the ac circuit with a load.

Let us do just that; terminate our filter with a 1000 Ω load (**Figure 3**), and see how it behaves (**Figure 4**).

The first thing we notice is that the in-band (low frequency) output is half of what it was to start with. R1 and the load resistor are essentially a voltage divider. This is known as insertion loss, and all passive filters have some degree of insertion loss, while simple *RC* filters have a lot of insertion loss. Also notice that the cutoff frequency at (0.707 × 50 V) has moved up considerably to over 3 kHz.

We can greatly decrease the insertion loss by replacing R1 with a reactance, in the case of **Figure 5**, an inductor L1. In fact, if L1 and C1 were perfect, the filter would have zero insertion loss. This will never be the case in real life, but insertion loss for a good filter can be a fraction of a decibel. The results are shown in **Figure 6**.

Notice that we now have no insertion loss in-band — again, assuming an ideal inductor for L1 and ideal capacitor for C2. Even with real-world components, the filter can have much less than 1 dB of insertion loss. Also notice that the slope of the cutoff is steeper, and the cutoff frequency is back down where it is supposed to be.

This is still not a great filter, but it's better. One can cascade a number of identical filters to obtain a much sharper cutoff, with the same cutoff frequency.

By simply swapping L1 and C1, we can change this filter response to a high-pass version seen in **Figure 7**. Actually, we don't swap the components directly, but rather we swap their reactance signs. In this example I have intentionally matched (almost) their reactances, but this won't always be the case. This is where *j* notation becomes really handy. In most simple high-pass or low-pass filter designs, one intentionally avoids *LC* resonance, as this results in ripple in the passband. In more complex filters, one must find a usable compromise between in-band ripple and cutoff sharpness. This is what makes advanced filter design so much "fun."

We have already described a band-pass filter, perhaps unknowingly as the single parallel tuned circuit. As useful as the single tuned circuit is, it doesn't have the best overall shape for a lot of communications applications, such as IF amplifiers. For our last example, I'd like to demonstrate a very very old band-pass type filter, which while less commonly used in modern amateur radio gear, is very interesting from a physics point of view. It also happens to be one of my favorite circuits. There's something oddly satisfying about it. Behold the duet of inductively coupled tuned tanks in **Figure 8**.

Here we have two identical tuned tank circuits, each tuned to 5 MHz, with K1 mutual coupling between inductors L1 and L2. The coupling factor 0.15, which will be between 0 and 1, where 1 is 100% mutual inductance. These two tank circuits are rather loosely coupled. The response is shown in **Figure 9**.

Now, the interesting thing is that with tighter coupling, the two "humps" will move farther apart, with a deeper valley in between. With less coupling, the humps will move closer together, and with extremely loose coupling, they will converge into a single hump, where the response is indistinguishable from a single tuned circuit. Like all multi-component filters, if you attempt to flatten the in-band response, you will sacrifice the steepness of the slopes of the upper and lower cutoff frequencies. All filter design involves compromises.

By the way, this phenomenon of coupled resonators is of great interest outside of electronics. Here is just one of many fascinating links on the subject: **https://www.exploratorium.edu/snacks/coupled-resonant-pendulums**.

For ages and ages, the double tuned transformer was used for intermediate stage amplifiers in radios, televisions, and radar sets. Most of us old-school radio repair guys spent a lot of time twiddling IF "cans," which was a major part of radio receiver tuning and alignment. This job was largely eradicated with the advent of fixed-tuned modular IF crystal or mechanical filters. The famous Collins mechanical filters actually were the equivalent of *multiple* coupled circuits, which created a lot more "humps" but with much shallower valleys between them, and with *much* steeper cutoff slopes.

In this essay, we haven't addressed the issue of phase shift through a filter, but it's a rich enough topic to justify an entire essay by itself; and that's just what we'll do next time.

Until then, keep those soldering irons hot! — *73, Eric*.

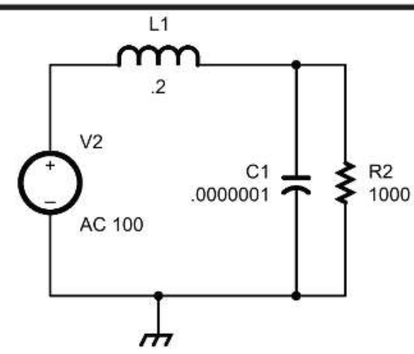


Figure 5 — Output with the resistor R1 replaced by inductor L1:

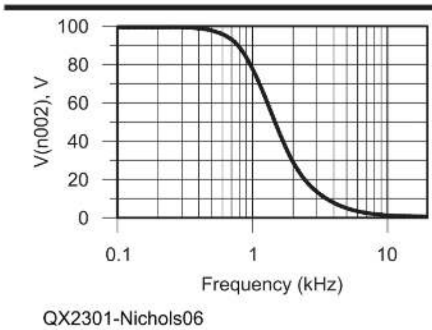


Figure 6 — Response of he *LC* filter.

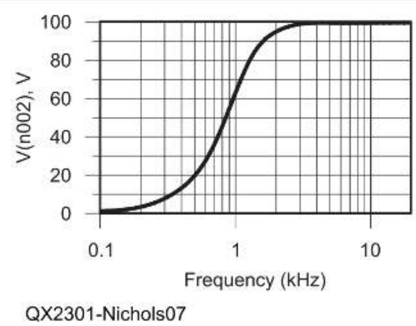


Figure 7 — A high-pass filter response.

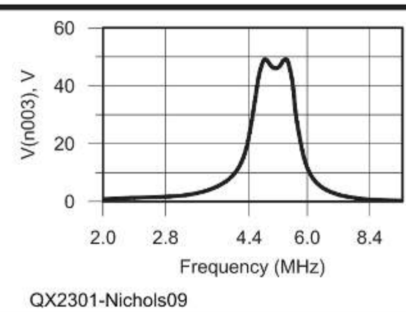


Figure 9 — Response of two loosely coupled tank circuits.

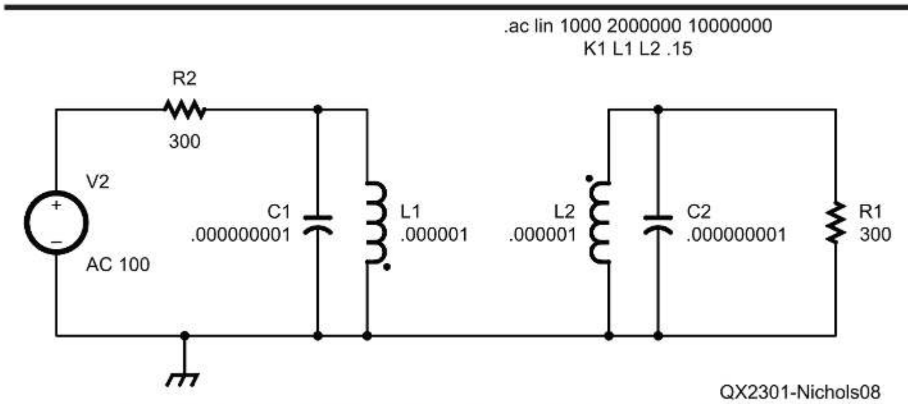


Figure 8 — The two tuned circuits are coupled by mutual inductance.

2022 *QEX* Index

Features

About the Cover

Announcements

Errata/Feedback

UNITED STATES POSTAL SERVICE® Statement of Ownership, Management, and Circulation (All Periodicals Publications Except Requester Publications)

1. Publication Title	2. Publication Number	3. Filing Date
QEX	0 8 8 6 – 8 0 9 3	November 1, 2022
4. Issue Frequency	**5. Number of Issues Published Annually**	**6. Annual Subscription Price**
Bi-Monthly: Jan/Mar/May/July/Sept/Nov	6	$29.00

7. Complete Mailing Address of Known Office of Publication (*Not printer*) (*Street, city, county, state, and ZIP+4®*)

225 Main Street, Newington, Hartford County, CT 06111-1400

Contact Person: Yvette Vinci

Telephone (*Include area code*): 860.594.0257

8. Complete Mailing Address of Headquarters or General Business Office of Publisher (*Not printer*)

225 Main Street, Newington, CT 06111-1400

9. Full Names and Complete Mailing Addresses of Publisher, Editor, and Managing Editor (*Do not leave blank*)

Publisher (*Name and complete mailing address*)

American Radio Relay League, Inc., 225 Main Street, Newington, CT 06111-1400

Editor (*Name and complete mailing address*)

Kazimierz "Kai" Siwiak, 225 Main Street, Newington, CT 06111-1400

Managing Editor (*Name and complete mailing address*)

Kazimierz "Kai" Siwiak, 225 Main Street, Newington, CT 06111-1400

10. Owner (*Do not leave blank. If the publication is owned by a corporation, give the name and address of the corporation immediately followed by the names and addresses of all stockholders owning or holding 1 percent or more of the total amount of stock. If not owned by a corporation, give the names and addresses of the individual owners. If owned by a partnership or other unincorporated firm, give its name and address as well as those of each individual owner. If the publication is published by a nonprofit organization, give its name and address.*)

Full Name	Complete Mailing Address
American Radio Relay League, Inc	225 Main Street, Newington, CT 06111-1400

11. Known Bondholders, Mortgagees, and Other Security Holders Owning or Holding 1 Percent or More of Total Amount of Bonds, Mortgages, or Other Securities. If none, check box → ☒ None

Full Name	Complete Mailing Address

12. Tax Status (*For completion by nonprofit organizations authorized to mail at nonprofit rates*) (*Check one*)
The purpose, function, and nonprofit status of this organization and the exempt status for federal income tax purposes:

☒ Has Not Changed During Preceding 12 Months

☐ Has Changed During Preceding 12 Months (*Publisher must submit explanation of change with this statement*)

13. Publication Title: QEX

14. Issue Date for Circulation Data Below: Sept/Oct 21-July/Aug 22 | Sept/Oct 2022

15. Extent and Nature of Circulation			Average No. Copies Each Issue During Preceding 12 Months	No. Copies of Single Issue Published Nearest to Filing Date
a. Total Number of Copies (*Net press run*)			5467	5400
b. Paid Circulation (*By Mail and Outside the Mail*)	(1)	Mailed Outside-County Paid Subscriptions Stated on PS Form 3541 (Include paid distribution above nominal rate, advertiser's proof copies, and exchange copies)	3062	3025
	(2)	Mailed In-County Paid Subscriptions Stated on PS Form 3541 (*Include paid distribution above nominal rate, advertiser's proof copies, and exchange copies*)	0	0
	(3)	Paid Distribution Outside the Mails Including Sales Through Dealers and Carriers, Street Vendors, Counter Sales, and Other Paid Distribution Outside USPS®	768	761
	(4)	Paid Distribution by Other Classes of Mail Through the USPS (e.g., First-Class Mail®)	360	376
c. Total Paid Distribution [*Sum of 15b (1), (2), (3), and (4)*] ▶			4190	4162
d. Free or Nominal Rate Distribution (*By Mail and Outside the Mail*)	(1)	Free or Nominal Rate Outside-County Copies included on PS Form 3541	52	51
	(2)	Free or Nominal Rate In-County Copies Included on PS Form 3541	0	0
	(3)	Free or Nominal Rate Copies Mailed at Other Classes Through the USPS (e.g., First-Class Mail)	56	36
	(4)	Free or Nominal Rate Distribution Outside the Mail (*Carriers or other means*)	82	82
e. Total Free or Nominal Rate Distribution (*Sum of 15d (1), (2), (3) and (4)*)			189	199
f. Total Distribution (*Sum of 15c and 15e*) ▶			4379	4361
g. Copies not Distributed (*See Instructions to Publishers #4 (page #3)*) ▶			1087	1039
h. Total (*Sum of 15f and g*)			5467	5400
i. Percent Paid (*15c divided by 15f times 100*) ▶			95.68%	95.44%

* If you are claiming electronic copies, go to line 16 on page 3. If you are not claiming electronic copies, skip to line 17 on page 3.

UNITED STATES POSTAL SERVICE® Statement of Ownership, Management, and Circulation (All Periodicals Publications Except Requester Publications)

16. Electronic Copy Circulation N/A	Average No. Copies Each Issue During Preceding 12 Months	No. Copies of Single Issue Published Nearest to Filing Date
a. Paid Electronic Copies ▶		
b. Total Paid Print Copies (Line 15c) + Paid Electronic Copies (Line 16a) ▶		
c. Total Print Distribution (Line 15f) + Paid Electronic Copies (Line 16a) ▶		
d. Percent Paid (Both Print & Electronic Copies) (16b divided by 16c × 100) ▶		

☒ **I certify that 50% of all my distributed copies (electronic and print) are paid above a nominal price.**

17. Publication of Statement of Ownership

☒ If the publication is a general publication, publication of this statement is required. ☐ Publication not required.

Will be printed in the January/February 2023 issue of this publication.

18. Signature and Title of Editor, Publisher, Business Manager, or Owner

Date: 11/4/22

I certify that all information furnished on this form is true and complete. I understand that anyone who furnishes false or misleading information on this form or who omits material or information requested on the form may be subject to criminal sanctions (including fines and imprisonment) and/or civil sanctions (including civil penalties).