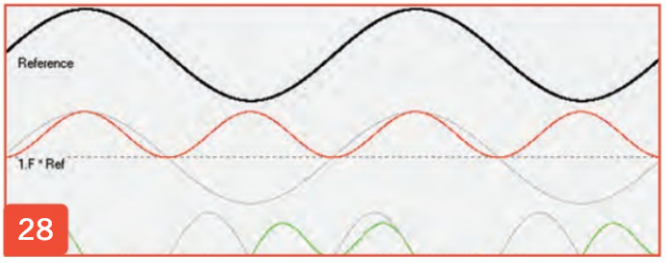
Contents

May 2024

Regulars

News and Reports

Reviews



Features

Technical Features

Kenwood TH-D75 review
Cover design by Kevin Williams, M6CYB

RadCom THE RADIO SOCIETY OF GREAT BRITAIN'S MEMBERS' MAGAZINE

Managing Editor: Edward O'Neill, M0TZX, edward.oneill@rsgb.org.uk
Technical Editor: Peter Duffett-Smith, GM3XJE
Layout and Design: Kevin Williams, M6CYB, kevin.williams@rsgb.org.uk

All contributions and correspondence concerning *RadCom* should be emailed to: radcom@rsgb.org.uk. Alternatively by post to *RadCom* Editor, 3 Abbey Court, Fraser Road, Priory Business Park, Bedford MK44 3WH Phone 01234 832 700.

RadCom is published by the Radio Society of Great Britain as its official journal and is sent free and post paid to all Members of the Society. **The June 2024 edition of *RadCom* is expected to arrive with most Members by 17 May 2024** although this can take up to a week longer in some cases; international deliveries can take longer still.

The online *RadCom* is at www.rsgb.org/radcom
RadCom Plus is available to RSGB Members online at www.rsgb.org/radcom-plus
RadCom Basics for Members new to the hobby can be found at www.rsgb.org/radcom-basics
Abbreviations and acronyms we use are listed at http://tinyurl.com/RC-acronyms

RADCOM (ISSN No: 1367-1499) is published monthly by the Radio Society of Great Britain and is distributed in the USA by RRD/Spatial, 1250 Valley Brook Ave, Lyndhurst NJ 07071. Periodicals postage pending paid at So Hackensack NJ. POSTMASTER: end address changes to RADCOM c/o RRD, 1250 Valley Brook Ave, Lyndhurst NJ 07071

www.facebook.com/theRSGB
Twitter @theRSGB

radcom@rsgb.org.uk

New Products

The Quadra PC bundle featuring the HamClock

This product gives you everything you need to run HamClock (3.04 currently, but will always auto-update to the latest version) on a 1080p or higher-resolution TV or monitor. It includes the Quadra PC, a mini wireless keyboard/trackpad, HDMI cable, ham-friendly USB power adapter, custom stand, a quiet fan with a short cable, and a USB hub for a neat, small footprint and cooler-running installation.

On first boot, the inovato Quadra PC will ask if you want to start up HamClock on boot, and if you'd like it to run full in screen (no black borders on the sides). It will do all the configuration for you so no need to create or edit startup files, change resolutions, etc.

The HamClock Setup automatically enables HamClock's 'Live Spots' feature so you can see where bands are currently active. All you have to do is enter your callsign and location on the first run and everything else will just work. Once set up, you can view your HamClock in a browser on any PC, tablet, phone or smart TV in your local network.

All of the standard Ham software is also installed, so simply pressing ALT-F11 will take HamClock out of full-screen mode so you can run other apps like WSJT-X, flDigi, JS8Call, or Pat Winlink.The Ham apps installer is always ready to install more, or update already installed apps.

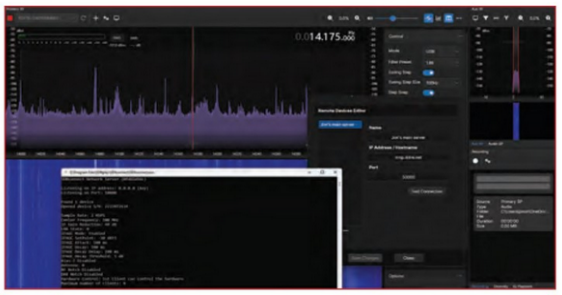
This product is currently available from moonrakeronline.com for £69.95.

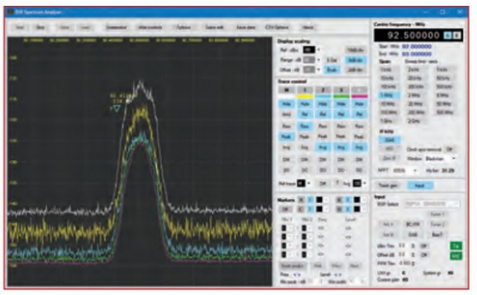
More software news from SDRplay

SDRplay's latest multi-platform software (SDRconnect) now includes an integral server which allows multiple users to access a single RSP radio receiver. This is proving very handy for people with good antennas and QRM-free locations to share their SDR hardware over the internet with their friends. In the past, this meant only friends with Windows PCs – now it can be anyone with a computer. Windows, MacOS or Linux/Raspberry Pi are all fully supported with both client and server versions of the SDR software. This also allows radio clubs to provide receiver access to would-be radio amateurs or members who can no longer maintain their own receiving station. Whoever connects first gets full control of the radio. Additional connections have to tune within the slice of spectrum selected which will typically be limited to one or two amateur bands. The owner has options to control this as desired. There are lots of new demonstration videos which can be found on sdrplay.com/sdrconnectvideoguides

SDRplay also reported that Australian spectrum analyser software developer Steve Andrew has updated his popular software to run on the latest RSP1B device. He has also made some significant improvements to the software and documentation. This includes major changes to his companion (Arduino-based) tracking generator. To get the latest news on this and to read the documentation, go to sdrplay.com/spectrum-analyser

The UK manufactured RSP family of SDR receivers cover from 1kHz to 2GHz with no gaps. They range in price from around £110 to £260 and are available directly from SDRplay Ltd (sdrplay.com), or from Martin Lynch & Sons, Moonraker, Nevada, Radioworld, SDR-Kits and Waters & Stanton.





CSN antenna tracker with GPS and stand

This is a self-contained antenna rotator and radio controller. It natively controls Icom radios, Yaesu rotators, and can interface with PSTRotator. This unit is a kit and requires some assembly.

Getting an antenna rotator to work with a computer can be an expensive and complicated task. The S.A.T. does away with the confusion – just attach it to a rotator and a radio. All you need is a web browser to control it!

This product is available from hamradio.co.uk for £279.

Antennas

When in the 1960s I lived on Ascension Island (ZD8) in the mid-Atlantic Ocean, I had a V-beam antenna of 100m wires that radiated over the sea towards the UK. Despite changing propagation conditions, I made a Q5 contact with a G friend on 14MHz every morning before we each went to work.

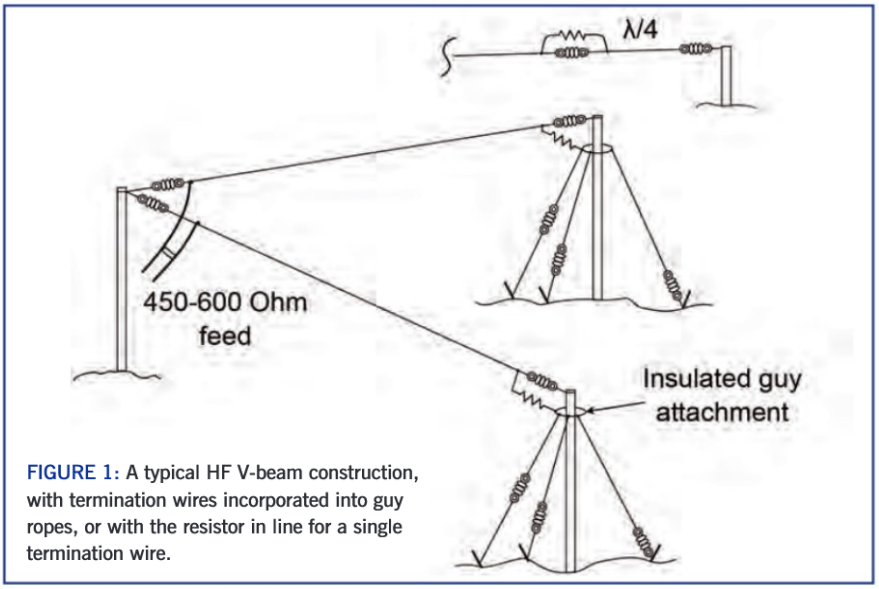


FIGURE 1: A typical HF V-beam construction, with termination wires incorporated into guy ropes, or with the resistor in line for a single termination wire.

Introduction

On another side of the island, US radio amateurs had a V-beam pointing to the USA. I used mine to great effect, and achieved the highest score in an RSGB 21/28MHz contest. Indeed, it was difficult to convince some UK radio amateurs that I was not their local pirate! The wire V-beam antenna, as in **Figure 1**, is an easily-constructed multiband high-gain HF antenna for use on a fixed point-to-point path, or in a contest where a specific country must be contacted. It is capable of producing a narrow, horizontally-polarised, beam of radiation, as may be seen in **Figure 2**. However, it does require a lot of land, and may seem impractical for most of us living, as we do, in pint-sized plots. Nevertheless, I thought it would be of interest to discuss its principles of operation, and to describe how to build one for all the farmers out there.

How the beam is formed

When a radiating horizontal wire is lengthened beyond a half wavelength, the single broadside radiation lobe, that surrounds it like a doughnut, splits into multiple concentric cone-like lobes around the wire. The closest cone to the wire intensifies as the others diminish. Polarisation of the radiated power changes from vertical, above and below the wire, to horizontal at the sides, and with mixed polarisation in between. When two such wires are formed into a V with angle θ between them, driven 180° out of phase and terminated with resistance, most of the vertical radiation components cancel, whilst the horizontal radiation components add to form a uni-directional horizontally-polarised beam that is reinforced by ground reflection, as shown in **Figure 3**. Relatively-strong off-beam horizontally-polarised minor lobes are formed by the sections of cones that are outside the V at approximately $\pm\theta$° off the main beam. These may be seen in the plot of **Figure 4** where the wires are each four wave lengths in length. The angle θ gets narrower as the frequency, or the wire length, are increased and may, for example, be less than 30° for wires ten wave lengths long, and more than 90° for wires of two wave lengths. The corresponding elevation take-off angles are 9° and 30° for wires suspended at typically-low heights above the ground.

Variations on a V-beam

The longer the wires, the less the height above ground required for a particular elevation take-off angle. Ultimately, when they are very long and very low, vertical polarisation and ground losses make for low-efficiency Beverage receiving antennas. Gain may be increased by extending the wires of a V-beam into an image V, forming a rhombus shape, that has a single terminating resistor at the point of the rhombus. The rhombic antenna so formed has more gain, and similar wideband characteristics to a V-beam. More gain may also be achieved, at a particular frequency, by placing a second identical V antenna with its feed-point a quarter of a wave length in front of the first one, and driving it with currents lagging by 90° with respect to those in the rear antenna. Sometimes, the wires of a V-beam are sloped down to the ground from the feed-point, with termination resistors connected to the ground itself. This has much less gain than a horizontal V-beam of the same length. A rotatable V-beam for reception has been formed from multiple sloping terminated wires, radiating from a central feed-point, adjacent pairs being selectable for each azimuth angle by a rotary double-pole motorised switch.

Construction

The V-beam has the advantage of not needing to be erected at great height to achieve the low elevation angles necessary for long-distance HF communication. **Figure 5** shows examples of how the take-off elevation from a V-beam is dependent on wire length at 14MHz, when compared with a horizontal dipole antenna at the same height above the ground. Wires of length 33m (108ft) at 10m height result in useful elevation angles from 14MHz to 30MHz, and longer wires that make the cone narrower will result in even lower take-off elevation angles with more gain for the same height at these, and lower, frequencies. This is consistent with how long-distance optimum take-off elevations naturally increase at lower frequencies. V-beams become most effective when the wires are at least three wavelengths long, 250m at 3.5MHz for example, and on such lower frequencies need considerable wire tensions to prevent them sagging. They may need intermediate supports for wires of copper-weld, galvanised or stainless steel, with final tension at ground level maintained using a back-stay turnbuckle.

Drive impedance

Low VSWR is achievable over at least a 3 to 1 frequency range on a 450Ω ribbon feeder, and even lower on a 600Ω feeder, or on a coaxial cable attached via an unbalanced-to-balanced 12:1 impedance-ratio transformer, without the wire lengths being critical. A tuner was not required when I used a 50-to-600Ω (12:1) wideband

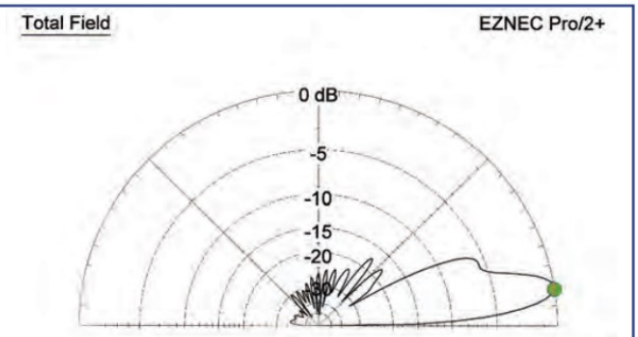


FIGURE 2: A simulated elevation beam plot of an eight-wavelength HF V-beam antenna.

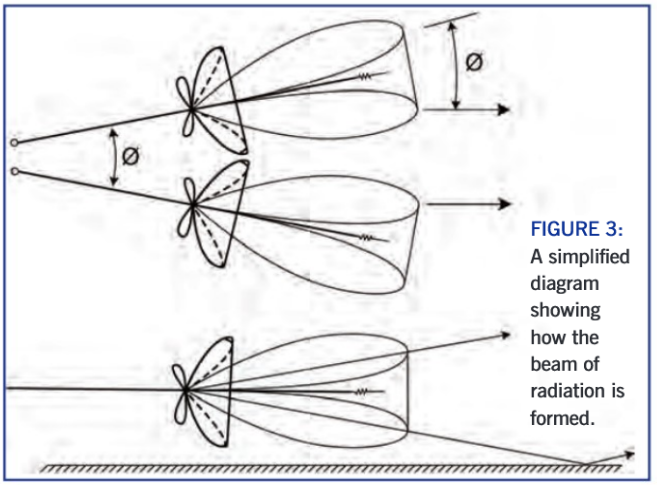


FIGURE 3: A simplified diagram showing how the beam of radiation is formed.

transformer at the feed point, because my National NCX5 valve radio had an adjustable pi output network. A solid-state radio may require a balanced tuner on lower frequencies.

Gain and directivity

100m wires at 10m height, with θ near 30°, provides good gain and appropriate take-off elevation angles for frequencies from 18MHz to 30MHz, whilst frequencies below 18MHz still benefit. These roughly coincide with respective periods of high- and low-sunspot activity. In either case, more gain, equivalent to doubling the power of the transmitter, occurs when the wires that form the horizontal V are not terminated. However, the antenna then has no more than a 3dB front-to-back (FB) ratio. This may be useful when the reciprocal path is closed, but will be a nuisance when both paths are open. FB ratios of more than 30dB are achievable, when resistors having the characteristic impedance of the wires are connected between the ends and a low impedance. This may be arranged by inserting the resistors a quarter-wave length from the remote end of the radiating wires. Ideally, a quarter wave termination wire for each band is required to provide a consistently high FB ratio. However, a good FB ratio may be achieved over a wide frequency range with three termination quarter-wave wires cut for the low, mid, and high frequencies of 7MHz (or 10MHz), 14MHz and 28MHz for example, depending on the length of the main wires. These may be incorporated into the guy ropes of the support poles as in Figure 1. I did this at my supports, which were each formed from one and a half 6m scaffold poles. I achieved a better than 10dB FB ratio over a 3:1 frequency range with a single termination wire of 5/8 of a wavelength at the highest working frequency. This helped to pull up the gain on low frequencies at the expense of the FB ratio, whilst providing high FB ratio on higher frequencies, where the echo effect of simultaneous long- and short-path propagation is more likely to be a nuisance. A half-wave single termination wire must be avoided, because it will effectively disconnect the resistor.

Table 1 illustrates the performance typically achievable when the wires of the V-beam are terminated with resistors and various odd lengths of single in-line termination wires. To put these figures into perspective: a typical 4-square array for 7MHz has about a 5dBi gain, an HF cubical quad antenna, or a full-size three-element 14MHz Yagi beam antenna at 20m height, has a 12dBi gain, a horizontal dipole for any frequency has a 7.2dBi gain, an inverted-V dipole has a 3.2dBi gain, and a vertical antenna for 14MHz over average ground has about 1dBi gain.

Switching the termination resistors

The values of the resistors should be the 500Ω characteristic impedance of long 2mm wires, and should be within ±5% for best FB ratio. Because the termination resistors need each to dissipate about 20% of the transmitter power, it is advantageous to remove

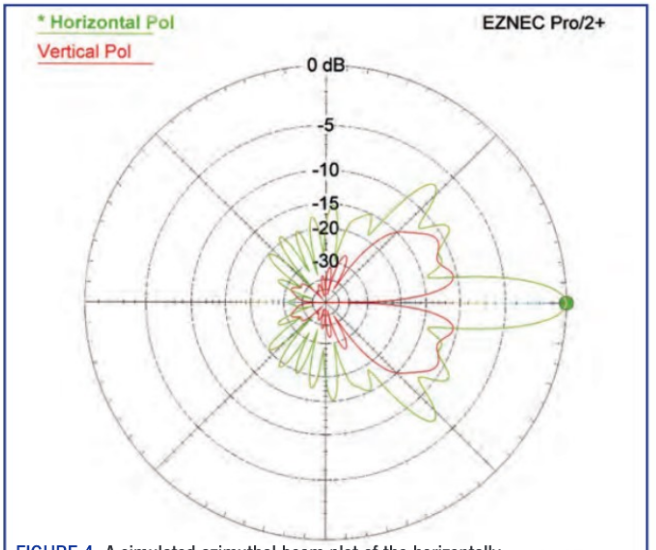


FIGURE 4. A simulated azimuthal beam plot of the horizontally-polarised (green) and vertically-polarised (red) components of radiation from the modelled four-wavelength example of an HF V-beam antenna.

them with T/R or PTT control when transmitting, and take advantage of the extra 2-3dB gain above the values in the tables below. In this case we can benefit from directivity when receiving, and use standard low-wattage non-inductive resistors, mounted across strain insulators. **Figure 6** shows some methods of 'removing' the resistors. Unless they are shorted whilst the wires are lowered, the sure way is to use relays by taping a fine ultra-violet-resistant enamelled wire to the main wires. Disconnecting the termination wires, unless they are in line with the main wires, is desirable because it eliminates their tendency to distort the radiation pattern. If high-voltage reed- or vacuum-relays are available, they may have their contacts connected in series with the resistor to disconnect both the resistor and the termination wires. Unless they have normally-closed contacts, the

Tony Preedy, G3LNP
g3lnp@yahoo.com

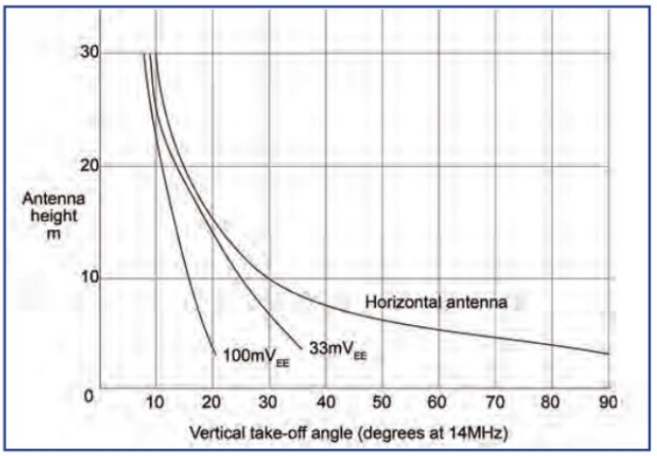


FIGURE 5: A comparison of take-off elevation angles of HF V-beams and horizontal dipole antennas, plotted as functions of height above ground.

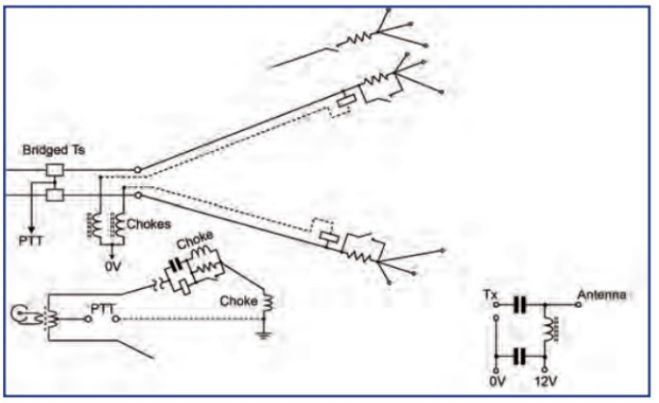


FIGURE 6: How to maximise radiation whilst transmitting by eliminating the termination resistors. This shows the circuit inside a typical bridged-T box, and alternative methods of feeding the PTT-derived relay supply, via a centre tap on the transformer or on a balanced-tuner's output winding.

PTT will now have to break the relay supply when transmitting. If a low-resistance ground connection can be provided at each end of the wires, it is practicable to pass the PTT current along the radiators, using chokes to both by-pass the resistors (to DC), and complete the ground connection with one of the termination wires at each pole. This may require a higher-than-normal relay voltage to overcome ground resistance and/or 50Hz AC to avoid electrolysis effects. Chokes, a pair of bridged-Ts for balance, or a transformer/tuner centre tap, may be used to inject the phantom control current.

If the FB ratio is not important, the resistors and termination wires may be omitted. When the V-beam is not terminated, a balanced tuner will almost certainly be required to cope with the increased VSWR. However, VSWR is much lower for an un-terminated V-beam when the length of the wires is a multiple of half-waves. Before switching them, I used carbon rods of 2cm diameter and 15cm length, with 200W PEP from my radio, after trying 240V 40W filament lamps. The problems with lamps include their change of resistance between transmitting and receiving, which is the opposite to what is desirable, and the obvious annoyance that their syllabic or Morse-code flashing may cause to neighbours and animals.

Modelling

The V-beam is easily modelled to fit an available site and required propagation path, as there are only two wires to work with. Start with the available distance along the centre line of the V in the required radiation direction, and the frequency. Then adjust the spacing, and the height of the supports, in the model for maximum gain at the required take-off elevation angle. For example, assume that we require a 13° take-off elevation angle, with maximum gain on 21.2MHz, from a site that is 60m long, together with an allowance for guying the supports. The resistors are inserted as loads at 3.53m from the ends of the wires. Starting with, say, 10m height and a 35° V angle, adjust the spacing of the wire ends for maximum gain. Then adjust the height for 13° take-off angle. In this case, the model should show height as 13m, gain 13dBi, FB 32dB, and maximum side lobes at -7dB. The distance between the end supports is 48m, making the V-angle $\theta = 2\tan^{-1}((48/2)/60) = 43°.6$. The length of each wire is, $60/\cos(43.6/2) = 64.6$m. Removing the resistors increases the gain to 15.2dBi with 2.5dB FB ratio. The polar plot for this antenna may be seen in Figure 4, and the performance on the bands 7MHz to 28MHz is shown in **Table 2**. However, Table 1 shows that we may also achieve 13° take-off angle from 100m wires, with a narrower V at 10m height, with about the same gain.

Table 1: The predicted performance of some V-beams of 2mm wires at 10m height over average ground with 500Ω terminations. The spacing of the remote ends, and the termination wire lengths, are as shown. Without the resistors, gain at the higher frequencies is about 3dB greater.

Frequency (MHz)	450W VSWR	Gain (dBi)	Elevation take-off (°)	FB ratio (dB)
33m wires at 62°, and termination wires of 3.3m				
14.2	2.5	7.5	25	10
18.1	1.7	9.6	20	14
21.2	1.8	10.6	17	28
24.9	1.6	11.6	15	20
28.5	1.2	12.2	13	18
50m wires at 50°, and termination wires of 4m				
14.2	1.6	8.1	22	19
18.1	1.9	10.2	18	26
21.2	2.3	11.3	16	18
24.9	2.6	12.4	14	13
28.5	2.8	13.2	12	9
100m wires at 33°, and termination wires of 3m				
7.1	3.8	4.4	28	5
10.1	2.2	6.8	22	13
14.2	1.6	9.5	17	19
18.1	1.0	11.4	12	18
21.2	1.7	13.0	13	19
24.9	1.9	13.7	11	19
28.5	2.0	14.5	10	20

Table 2: Calculated performance for the modelled example of 64.6m wires at 13m height, and a 44° angle with termination resistors at 3.53m from the ends of the wires. The 600Ω VSWR is in brackets.

Frequency (MHz)	450W VSWR	Gain (dBi)	Elevation take-off (°)	FB ratio (dB)
7.1	5.5 (4.1)	5.2	33	5
10.1	2.3 (2.0)	7.6	25	10
14.2	2.4 (1.8)	10.4	18	13
18.1	1.8 (1.4)	12.1	15	28
21.2	1.6 (1.3)	13.2	13	32
24.9	2.3 (1.8)	14.3	11	14
28.5	1.5 (1.6)	15.0	10	12

Review

Aziloop DF-72 VLF-HF multi-directional receive loop

The Aziloop DF-72 is a new product designed and sold by QuietRadio, run by Dave Evans, GW4GTE. Dave says that the Aziloop product development was started more than five years ago, the original requirement being for a multi-purpose antenna that could be remotely controlled.

Background

Nothing suitable existed, and so Aziloop was born (Azi is short for azimuth), with the DF-72 being the culmination of this long development cycle. The receive loop is highly innovative in that it can function in two modes, changing at the click of a mouse button.

The first mode is 'K9AY mode', where the Aziloop operates as a directional K9AY terminated loop giving a uni-directional cardioid pattern **[1]**. The second is 'loop mode', which produces a classic small-loop figure-of-eight bi-directional pattern at low angles, and an omni-directional pattern at higher angles.

The receive antenna is unique in using Dave's 'Stepped-Azimuth™' technology to produce up to 72 uni-directional headings in K9AY mode, or 36 bi-directional headings in loop mode, from a pair of orthogonal loops. The result is a choice of 108 heading and mode combinations, with headings controllable in five-degree increments. To be clear, the antenna does not move; rotation is achieved electronically, the whole antenna being controlled from a PC-based application. Microsoft Windows 10 and 11 are officially supported; however Windows 7 has been tested and found to work too.

FIGURE 1: The DF-X common interface unit (CIU).

So what do you get for your money?

The Aziloop DF-72 consists of a DF-X common interface unit (CIU – see **Figure 1**) and a DF-72 loop control unit (LCU – see **Figure 2**), together with a DC power lead, USB cable, 3.5mm-to-phone AUX cable, and two SMA to BNC adapters.

The DF-X CIU is a small metal box that has a dual role. The first is that it acts as a power supply for the antenna, feeding the power down the coaxial cable to the LCU. The second is that it also feeds the control signals to the loop, allowing you to select the mode, pre-amplifier status, K9AY impedance selection, and much more. The result is a tiny box that can sit somewhere in the shack, which is controlled locally by a USB or LAN connection to your PC. The DF-72 LCU is the loop controller itself, which is mounted outside, and has two wire loops attached (which you supply), one in a north-south orientation and the other east-west.

The LCU is housed in a high-quality IP67-rated waterproof enclosure made from acrylonitrile styrene acrylate (ASA) thermoplastic (as opposed to ABS) for excellent weather and UV resistance, zero corrosion, and minimal condensation. The lid of the enclosure is secured with six stainless-steel screws, which clamp it firmly in place via a waterproofing gasket. Cables are fed into the box via compression glands, and are clamped using rising-cage-type terminal blocks. The enclosure includes integrated pole mounts,

FIGURE 2: The DF-72 loop control unit (LCU).

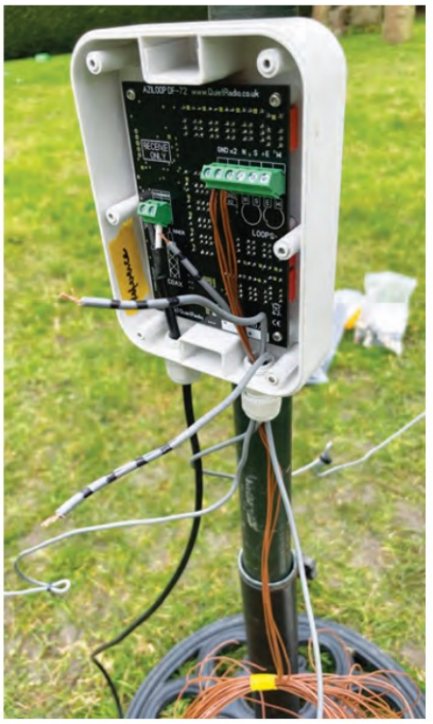


FIGURE 3: An internal view of the LCU.

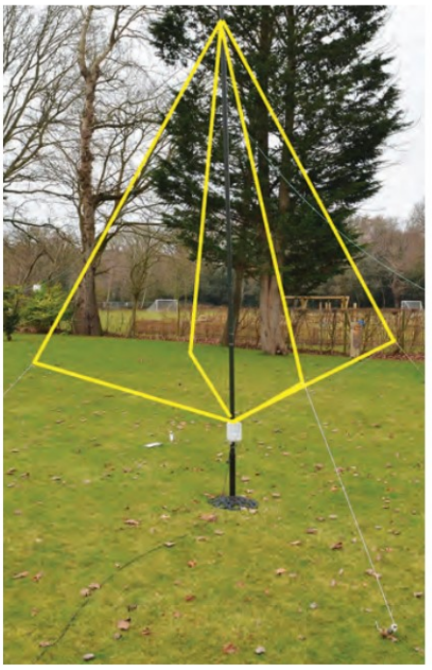


FIGURE 4: The completed assembly. The wires themselves are outlined in yellow to make them more visible.

allowing installation to be completed quickly and easily with just a couple of cable ties or jubilee clips (see **Figure 3** for an internal view). You have to supply a way of mounting the antenna. Chris, G0DWV, decided to use three fibreglass poles fitted into a cast-iron garden parasol support for our tests. This worked well, but you could also use a ground stake and the lower sections of a fibreglass fishing pole if you wished. Note that the pole must be non-metallic.

FIGURE 5: The computer application has a straightforward interface.

FIGURE 6: K9AY mode (uni-directional).

Assembly

Once you have cut two pieces of wire, typically each 9m in length, you can start to put the loop together. **Figure 4** shows the completed assembly. The wires themselves are outlined in yellow to make them more visible. I did not actually use yellow wires!

To connect the four loop cable ends to the LCU, you just have to take the cover off (six small crosshead screws), push the cables through the waterproof gland and connect them to the circuit board. A good tip is to mark the wires as north-south-east-west to make the job a little easier! Plastic cable ties are used to fasten the antenna to the pole. Once connected you then replace the waterproof lid, stake out the loop as shown, and connect up your coaxial cable.

You will also need to install four radials, which connect to the earth point in the LCU. Position them directly under the loop wires and extend them several feet beyond the footprint of each loop. This isn't critical, but keep to the same length for each radial. The instructions also have details of how to use ground stakes and elevated radials; you may need to experiment for the best results.

As I said earlier, everything is controlled through the co-ax, so you don't have to install a rotator or other control cables. That makes it easy to install and use.

Back in the shack, connect the coaxial cable to the CIU, connect that to a 13.8V supply via the supplied 2.1mm barrel connector, run a USB cable from the CIU to your PC and then run a piece of coaxial cable from the CIU to your receiver. The CIU also allows you to connect a PTT muting line, which removes power from the antenna when your PTT is pressed. One word of warning: if possible, use a receive-only input or run the coaxial cable to a separate receiver, rather than a transceiver, to avoid applying RF to the loop, which would probably destroy its delicate electronics.

If you have to connect the Aziloop directly to a transceiver's main antenna connector, be extremely careful. Remove the mic, turn the mic gain and RF level to zero, avoid AM or FM, and make sure the rig isn't on VOX. Be aware also that some transceivers can emit a high-power spike on transmit even on

Steve Nichols, G0KYA
steve@infotechcomms.co.uk

FIGURE 7: Loop mode (bi-directional).

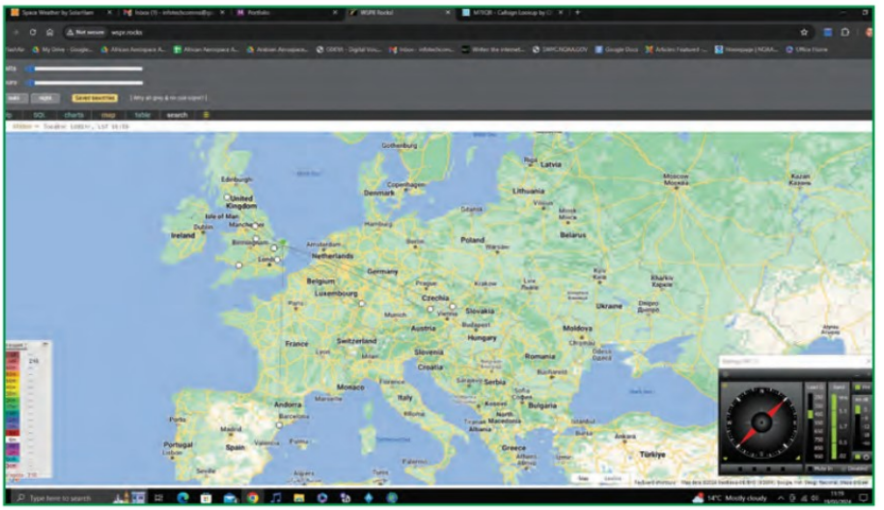


FIGURE 8: Impressive results using WSPR.

low power. Also, for locations with co-sited transmitters, place the loops as far away from hot antennas as possible, and make sure you use the Azuloop's built-in PTT muting function, especially if you run high power.

Now you just have to install the AziLoop software, get it to recognise the unit via the USB cable, and away you go. The current version of the software is 1.3.0, which now includes mouse-wheel tuning.

Running the Azuloop

The Azuloop software allows you to select classic loop or K9AY mode, five-position attenuator steps, an 18dB pre-amplifier, and much more (**Figure 5**). In K9AY mode (**Figure 6**) you can also select a different termination resistor value, adjustable in 50Ω increments from 250Ω to 950Ω. Ahead of the preamp are three selectable seven-pole filters, two low-pass and one high-pass. **Figure 7** shows the display in loop mode. The application displays a compass rose, within which you can change the loop direction with the mouse wheel or by clicking.

You can also configure the application to work over the internet using the CIU's RJ45 socket and built-in Ethernet server. The supplied instructions (which can be downloaded at **[2]**) run to 120 pages, so we have barely scratched the surface.

How does the Azuloop perform?

I started off by looking at LF, in particular the non-directional beacons (NDBs) around 350-450kHz. I soon spotted ONW (Antwerp, 355kHz), CWL (Cranwell 423kHz), and HEN (Henton, 434kHz). Later that evening I also spotted PLA (Pula, Croatia 352kHz). None of these was audible on a 132ft (40.2m) long-wire, showing how effective the low-noise characteristics of the antenna are. I think NDB hunters would be very pleased with the loop's performance.

Moving on to 472kHz, and using WSPR (see **Figure 8**), I soon had the loop receiving stations from around the UK and Europe. Leaving it running overnight, the best DX received was from EA3IHV at 835mi distance, OK0EMW at 754mi, and OE3EMC at 673mi. Most of these were inaudible at -20 to -29dB SNR. I think this antenna would be a valuable addition to those interested in LF.

Moving on to medium wave, and BBC Radio Wales on 882kHz sounded like a local station up here in Norfolk. Using the Azuloop's azimuth control, I could easily peak the signal or null it out. BBC Radio Scotland on 810kHz was equally strong, but I could also reverse the antenna direction and locate a Spanish MW station (COPE) on the same frequency with Radio Scotland becoming inaudible. Buoyed by those results, I checked 1130kHz, which is on the frequency for WBBR in New York, and by playing with the loop controls I was surprised to find that it was audible at 11pm in the middle of March. Nothing but noise could be heard on my long-wire antenna. The Azuloop is a very effective medium-wave antenna.

Moving to the 160m band, various German (DL) and Italian (I) stations could be heard on LSB around 11.30pm. By adjusting the direction of the loop, their signals could be reduced from S9 to S5 and from S5 to zero, showing how effective the loop is at nulling signals if needs be.

A quick check on 20m at 10.30pm on a different evening showed that, by orienting the loop in different directions, a number of different IBP beacons could be heard on 14100kHz, including 4U1UN (New York), YV5B (Venezuela) and 4X6TU (Israel). Given the time, this was quite impressive.

The benefit of having a receive-only antenna is often cited as the biggest improvement to making contacts on the low bands, especially 160 and 80m. When you install a beverage system or even a one-direction beverage or long wire, it is immediately apparent that you are hearing stations you wouldn't have heard before as they were probably in the noise of the Tx antenna, especially with verticals. The signal-to-noise ratio (SNR) is the key component in this, but also the arrival angle of the signals. However, you can't rotate a beverage or even a dipole to null or peak signals, and this is where the Azuloop comes into its own.

There were many times in the three months of G0DWV's testing that showed the Azuloop to be a highly-effective receive antenna, maybe not a replacement for long beverages, but a very strong complement to them, for as signals dipped on one they peaked on the other. If you have a small garden it will transform the lower bands and give other options on the higher bands. The directionality is very effective, and you can usually find a direction that will allow the contact to be made even with strong QRM, and even if not beaming in the correct direction.

In summary, the Azuloop is an innovative receive antenna system that does away with the need to use a rotator, and can be fed with a single piece of coaxial cable. It may suit short-wave listeners and radio amateurs who operate on the lower bands. It offers a low-noise solution to the reception of signals from VLF to HF, and can be quite addictive to play with. It is currently available at a special UK price of £399 plus UK delivery of £6. Its future list price will be £459. The Azuloop is shipped in a small F5 size (8" x 6" x 4") package. Our thanks to Dave, GW4GTE for the loan of the antenna. More details can be found at **[2]**.

References

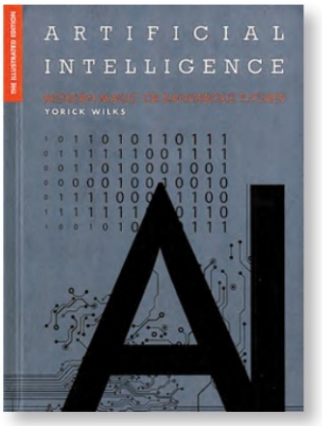
[1] https://www.aytechnologies.com/TechData/LoopInstall.htm
[2] https://www.quietradio.co.uk

Book Review

RadCom Team radcom@rsgb.org.uk

Artificial Intelligence

By Yorick Wilks



This books explores a controversial subject and is a novel way of introducing this to the readers. Artificial Intelligence (AI) has been portrayed as a somewhat dark art with sinister overtones, as seen in various films portraying the overthrow of man by machines.

The book begins by discussing the basic principles of AI and the implications of its use in society. This is a non-technical book and thus will appeal to many who have an interest in the subject but do not wish to immerse themselves immediately into the deep mathematical code required to make AI function.

Let us have a brief look at just a few of the questions raised by the book for discussion. Avatar design is a most contentious subject. The question that usually arises is: do we need to make AI in our own image and, if so, in order to be believable, should it be able to show emotional response? This then leads to the subject of avatars becoming companions and as such absorbing huge amounts of information about their owners. At this point ethics and legality enter the argument as, whilst information may possibly become the singular property of its owner, unlike the present world we live in, the companion and its store of information would be most useful in the event of a crime to the authorities etc. A complete change in thinking would be required to accommodate companion living.

Socialisation is another area discussed. Should we be interacting with AIs in the same way as we do with humans and, if so, what are the long-term implications for society?

As society grows older, could AIs be a repository or augmentation for memory? When memory begins to faulter could this be a way of providing reference and would it be treated with reverence after the death of its owner? The subsequent question would then be who would take custodianship of those recorded memories.

Reading the book brought to mind the Google app Alexa and Apple's Siri which are both AI based. Some raise questions about 'always on' listening devices which may be constantly gathering information about its users and storing it.

One interesting aspect of AI ethics explored in the book is the test called the 'trolly problem'. This involves a question which asks what you would do if you had a vehicle that was out of control. If it was going to kill someone, and you had a choice of killing a baby or a senior citizen, who would you choose? Humans from different cultures choose different answers. AI logic does not differentiate which is why they may become an option for use with future solders.

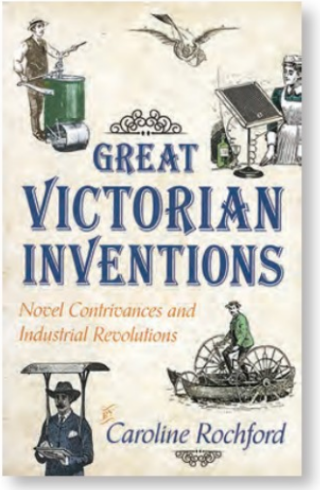
The biggest question that we all wish to know the answer to is whether or not AIs will become conscious or, to put it another way, self-aware. Well, you'll have to read the book to find out more about this most interesting of questions.

The book is certainly thought provoking, very informative and gives the reader a perspective that perhaps they had not considered previously.

Size 170x235mm, 256 pages, ISBN: 9781 7857 8993 9
Non Members: £20.00, RSGB Members: £15.99 (20% OFF)

Great Victorian Inventions

By Caroline Rochford



How you ever stopped to think about where most of our modern conveniences come from? Most of us would assume wrongly that they were very recent inventions when in fact a great many had their origins in the Victorian age. Literally hundreds of items we now take for granted were actually Victorian inventions that have been refined over the years and generations. For example, did you know that the manual carpet sweeper was one of them and not the result of a 1960s vacuum cleaner salesman's attempts to doorstep sell you something you never wanted? Other items include: the rowing machine, gymnastic treadmill, the diving helmet, the telegraph, solar-powered heating, and many more. In fact, virtually every facet of life in those times is covered within the text.

The book covers over 200 examples of Victorian inventions that were designed to enhance life for both the privileged and the people in service to them, as was the custom in those times.

Some of the items are as simple as the 'foot dustpan' to make carpet cleaning easier for the house staff. Usually, staff would have to work on bended knees which was an arduous task. The invention of the 'foot dustpan' eliminated this as, whilst standing, it could be controlled with one's foot whilst sweeping with a conventional broom.

An example from a completely different arena is the 'Hand Granade Fire Extinguisher' which was a corrugated glass bottle containing a chemical which when thrown into a fire and would instantly generate a large volume of carbon dioxide and extinguish the flames. Unfortunately, it was later discovered that the chemical used, namely Carbon Tetrachloride, was highly toxic to people and animals.

The book has been divided into various subject areas: The Home, Leisure, Fashion, Art and Design, Education, Work and Industry, Electrical inventions, Lights Camera Action, Health and Safety Victorian style, Science and Nature, Technology and Communication, Travel and Transport. The reason for this is clearly due to the sheer number of inventions covered.

This is a truly fascinating book and once picked up was hard to put down. It makes one realise just how much the Victorians contributed to our way of life today.

Size 124x198mm, 288 pages, ISBN: 9781 4456 3617 7
Non Members: £10.99, RSGB Members: £8.79 (20% OFF)

Technical

What actually goes on inside an FFT?

Introduction

In the previous article in this series, we looked at digital filtering and showed how a buffer-full of input samples was manipulated to give a filtered version of the input waveform. Here, we look at another process that uses broadly similar 'hardware', but for a completely different purpose. This is the 'Fourier transform', used to generate frequency data from an input time waveform, and its high-speed computer-optimised variant, the 'fast Fourier transform' (FFT).

In the same way as is done for digital filtering, an input analogue waveform is sampled, using an analogue-to-digital converter (ADC) running at a sampling rate of F_s, into a time-ordered series of digital samples, and these are stored in a serial buffer. The buffer therefore contains the last N samples at any one time. For reasons that will be explained later, the buffer length is usually an exact power of two, like 512 or 65536, and is generally much longer than one used for digital filtering. Sometimes the buffer length can extend to over one million samples. The contents of the buffer are used in the FFT process. As in the case of a digital filter, each historical sample is multiplied by a coefficient and summed in a multiply-and-accumulate (MAC) process. But that is where the similarity ends.

The basic process

We want to extract frequency data from a time series of input samples. We do this by multiplying a block of samples of the unknown input by a series of sampled sine waves, one sine wave for each frequency component we wish to test for. To explain how this works, we need to look at the mathematics of multiplying two sine waves together. A sine wave of frequency F_1 can be expressed mathematically as $\sin(2\pi F_1 t)$, where t represents the time. If we have two sine waves, one of frequency F_1 and the other of frequency F_2, then the result, Y, of multiplying them can be written as

$$2Y = 2\sin(2\pi F_1 t)\sin(2\pi F_2 t),$$

and this can be re-expressed as the sum of two cosines

$$2Y = \cos(2\pi[F_1 + F_2]t) + \cos(2\pi[F_1 - F_2]t).$$

Now a cosine is the same as a sine but with a phase shift of 90°, so the output just contains two components with frequencies equal to $F_1 + F_2$ and $F_1 - F_2$. This is our standard frequency mixer process, with the well-known sum and difference terms shown mathematically.

Consider what happens if F_1 and F_2 are the same frequency, ie $F_1 = F_2$, (and to simplify things we will, for now, assume that the waves are in phase with each other). One output term is equal to $2F_1$, the second harmonic, and the other is equal to zero frequency, ie DC. In the FFT processing, the results are averaged over time, so the second-harmonic term disappears, since the average of a sine or cosine wave is zero. However, the DC term is a constant since $\cos(0) = 1$, and this does not average to zero. Thus, by multiplying the input waveform by a sine wave of frequency F_1 and then averaging the result, we pick out just the component in the input waveform which has a frequency of F_1.

By now it should be becoming apparent how the Fourier transform works. We multiply our input waveform by a sine wave of a given frequency, F, and average the result. We get a constant output if there is a component of the input waveform at frequency F, and zero output if there isn't. By stepping F over a range of frequencies, and repeatedly multiplying and averaging for each frequency, we map out the individual frequency components in the input waveform.

Figure 1 shows this graphically for the first five harmonic terms. The reference (shown in black at the top) is multiplied in turn with each of the light grey harmonics, to give the overlying coloured waveforms. It is clear that the average of the first one in red, where the test frequency is equal to the reference, averages to a positive value. The other traces, 2F, 3F etc, can be seen to average to zero.

Practical realities

In practice, we can't just 'multiply the input by a bunch of different reference sine waves'. We instead have to use a finite length block of input samples, and multiply these with a block of reference sine-wave samples. The block, or buffer, length can be anything we like but, because it is finite, it restricts the quality and accuracy of our results.

The lowest, or base, frequency we can derive from a block, or buffer-full of samples, is the case where the buffer contains exactly one complete cycle of samples of the sine wave of the reference. The next term is where the buffer contains two complete cycles, then three complete cycles then four and so on, up to the Nyquist limit where the reference is just a series of alternating plus and minus terms in each buffer location, equal to the base frequency times half the buffer length. This base frequency, equivalent to one cycle over the buffer length, is the resolution with which we can measure the frequencies in the calculated spectrum. Each term in the frequency data represents the amplitude of each multiple of this frequency up to half the sampling rate.

And this is the basis of the Fourier transform process; we fill a buffer of length N with a series of input samples, and repeatedly multiply each stored input sample with a series of reference sampled sine waves, starting at 1 and going to the N/2 harmonic, accumulating the results for each set of multiplications of each frequency component. The resulting single accumulated sum for each frequency gives the amplitude of the frequency component at that frequency. So the series of N/2 MAC results now corresponds to the amplitudes of each component of our input waveform. We have generated the spectrum of our input data, just what we wanted to do, up to a limit of half the buffer length multiplied by the base frequency. As this is being done on a discrete set of samples, it is usually referred to as the discrete Fourier transform, or DFT. Each frequency buffer location is termed a 'bin', as in dustbin, as each one contains a separate item of information, ie the amplitude of each frequency component. The width of each bin, or the 'bin size', is F_s / N, and this is the resolution to which we can measure each frequency component.

Rounding and in-between frequencies, and phase

What happens if a component of the input is not an exact multiple of the buffer length, but falls somewhere between one of the two reference frequencies, say at 3.5 times the base frequency? In that case, both the third and fourth reference terms will provide a non-zero averaged result. This will not be as much as if the exact frequency were to be hit, but some result will appear for both 3rd and 4th reference terms. In this case, other higher reference frequencies may also give some MAC output, but this will progressively reduce in amplitude as the harmonic number goes up. The situation can be ameliorated by 'windowing' the input data as we'll see later.

So far, we have assumed that the input frequency component was exactly in phase with the reference, so that $\cos(2\pi[F1 - F2]$

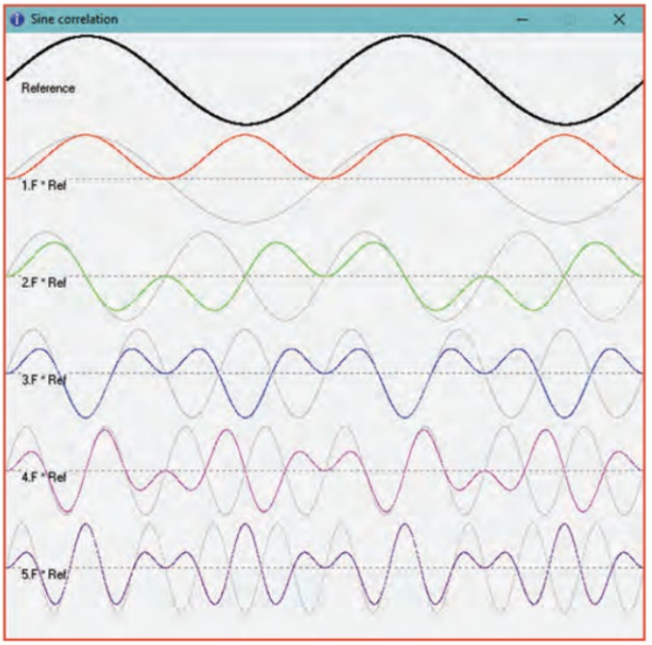


FIGURE 1: **The correlation of sine waves. The top trace in black is multiplied by 1, 2, 3, 4 and 5 times its frequency, each shown in grey, then averaged over the entire width. Note how only the plot in red, 1F, gives a finite average value. All the others average to zero.**

t)) = 1 for F1 = F2, but this will hardly ever be the case. In practice, we use both cosine and sine terms of the reference frequencies, and calculate a pair of results with a 90° phase difference between them. Let's assume our input is 90° out of phase with the original reference, ie cos(2πFt). Now 2cos(2πFt)sin(2πFt) = 0. But by simultaneously doing two MAC processes on each input sample, with cos and sin terms, or 'real' and 'imaginary' in the parlance of complex algebra, or 'I' (for in-phase) and 'Q' (for quadrature-phase) as electronic engineers call them, we generate a pair of results, sin(2πF1t).sin(2πF2t) and cos(2πF1t).cos(2πF2t) (the latter also gives a constant averaged output when the frequencies are the same). The two MAC results, when added together, then give the amplitude of the input component, whatever its phase. As the results are 90° apart, vector addition is needed to add the two numbers using Pythagoras, so the amplitude is √(I2 + Q2). There is also the added advantage that, by comparing the amplitudes of the cosine and sine terms, we can calculate the phase. This is of little value if all we want is a spectrum trace, or waterfall display of the input, but is vital when the Fourier transform is used in the decoding of multi-carrier waveforms, as used for most modern digital signalling like terrestrial TV and DAB broadcasting, and 4G/5G mobile phone signals.

This use of paired cos/sin multiplications is analogous to the I/Q channels we are familiar with in so many other aspects of DSP, and we'll see later that feeding I/Q data into the FFT process actually helps.

Putting in some numbers

So far, we have referred to a 'base frequency', or lowest spectral component, that can be detected, equivalent to one complete sinusoidal cycle over the buffer length. In reality what we are doing is feeding in a series of samples at a sampling rate F_s. When the buffer is filled, its complete length is then equivalent to a 'base frequency' of F_s/N. Thus, if we are sampling at 12kHz and have a 1024-sample buffer, the frequency components are evaluated at multiples of 12000/1024 = 11.72Hz. A longer buffer length of 2^{20}, or 1048576 cells, gives us a resolution of 0.0114Hz, but requires over a million complex MAC operations to get there, just for one frequency component, or 'bin'.

If we bear in mind that N separate MAC operations are needed for each frequency component, and we are evaluating N/2 frequency terms (both I and Q at each frequency), it becomes evident that we need $N^2/2$ MAC operations to evaluate the complete spectrum using the DFT on each block of data. For a buffer length of 1024, that is about half a million complex I and Q MAC operations for every set of results. At 11.72Hz resolution, a new spectrum will be generated every 85 milliseconds, needing a processor capable of doing 500000/.085 or about 6 billion MAC operations per second just for this relatively-low resolution answer; this is barely feasible even with the fastest processing hardware. Doing this for the million-sample buffer is totally unrealistic.

It looks as if the DFT process just described is totally impractical for real-time use, other than for low-resolution frequency analysis run on fast processing hardware. But fortunately there is a solution, and it was discovered long before the advent of digital electronics, around the time of the sea battle of Trafalgar, by Carl Friedrich Gauss (see Wikipedia).

The fast Fourier transform (FFT)

In 1965, Messrs. Cooley and Tukey popularised Gauss's method whereby the matrix of multiplications needed for the DFT could be greatly reduced. They made use of the fact that, for multiple sine values, many numbers repeat, so a multiplication at one point of the input waveform by one reference sine value is equal to another multiplication at a different point on another. Their algorithm optimised this process, doing just exactly the number of MAC commands needed for the complete DFT. This reduced the number of commands needed from $N^2/2$ to $N.\log_2(N)$, which means that, for the 1024-point example described above, the number of operations reduces from around 500000 to 10240, fifty times faster. For longer buffer lengths, the increase is even more dramatic. A buffer size of 1 million points, instead of needing something like the completely impractical 10^{12} MAC operations which would take several hours to compute on a GHz-clocked processor, is now reduced to 20 million operations, which it can easily manage in the time it takes for the buffer to refill with the next set of data.

The FFT process is quite complex to visualise, and will not be described in fine detail here. In essence, input samples are reordered and broken down into a tree structure of multiple small DFTs of just two samples at a time, whose results are added together. This 2x2 minimal DFT is often referred to as a 'butterfly' because of its appearance when shown graphically, and involves no actual multiplication as the coefficients, after breaking things down this far, are either 1 or -1. The number of actual multiplications for the whole FFT is now vastly reduced. Once the processing through the tree has finished, the samples have to be reordered, and this involves reversing the bits of the binary number representing each cell. Alternatively, the bit-reversal process can be done on the input samples before the tree of small MAC processes. The entire process is often done in-situ, using the same buffer memory as the stored samples. The buffer array enters the FFT process with time-domain data, and exits containing frequency-domain (or 'spectral') data. The need for bit-reversal, and the way the tree is broken down ultimately to a set of repeated 2x2 multiplications, is the reason that the buffer length has to be a power of two: 256, 1024, 65536 etc.

Filtrate
radioecm@googlemail.com

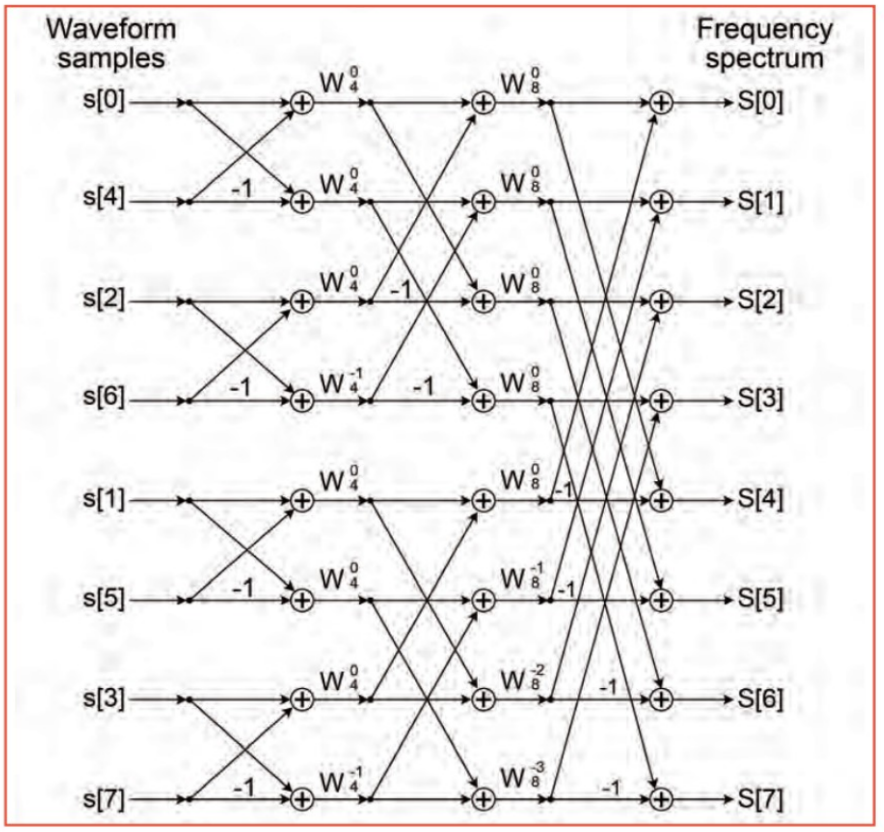


FIGURE 2: **Diagrammatic view of an 8-point FFT, showing how the re-ordered (bit reversed) input data is reduced to multiple two-point DFTs in a tree structure.**

Although precise details of the FFT are not described here, **Appendix 1** shows a listing of an FFT algorithm in one dialect of the BASIC language. The same buffer is used for input and output. Data enter and exit the routine in the array `dat()`, which contains alternate I and Q samples starting at an index of 1. For a simple serial-input stream of samples, every alternate location (I data) is filled with these, and the Q samples in-between are set to zero. The BASIC programme was derived from a FORTRAN process dating from the mid nineteen sixties, and the bit-reversal process carried out at the start may not be the most-efficient way of doing this. As bit reversal and reordering is just a small part of the processing overhead, no further optimisation was made in this part of the FFT routine.

Figure 2 shows a diagrammatic view of the process for an 8-point FFT, illustrating how samples are reordered, multiplied by the weightings (the sine and cosine values, labelled W_N) and summed in the circles with a plus sign. Note how the input sample numbers on the left have been re-ordered 0, 4, 2, 6, 1, 5, 3, 7. The binary equivalents of these are 000, 100, 010, 110, 001, 101, 011, 111, and it is clear that this is a bit-reversal of the original order 0, 1, 2, 3, 4, 5, 6, 7 or in binary 000, 001, 010, 011, 100, 101, 110, 111.

Windowing

The DFT or FFT process works on a fixed-length block of input samples, whilst the input data is of an arbitrary waveform shape. The start and finish values in the buffer will be different, whereas all the multiple sine and cosine reference frequencies assume a perfect start and finish. This causes a bit of 'leakage', or non-zero summations, where the result would otherwise sum to zero if start- and end-values had been the same. The effect in the frequency domain is to raise the noise floor and limit the dynamic range that can be detected. The solution is to multiply the set of input samples by a 'window' that reduces the start- and finish-samples to near zero, keeping those in the middle the same. The frequency resolution is then reduced, and two, or even three, bins can be affected by one frequency component. There is a compromise between dynamic range and effective frequency resolution (ie the number of bins affected), depending on the window type in use. Using no window, for example, offers the maximum frequency resolution of one bin, but limits the dynamic range potentially to 13dB. A Hamming window is usually taken as a good compromise, and allows around 40dB dynamic range with a resolution of about two bins. A more-extreme window is the Blackman-Harris window that offers 90dB dynamic range, but a frequency resolution of three bins. This is why most spectral-analysis and digital-receiver software allows the user to choose the window type, to optimise frequency spectral data or dynamic range.

Complex, or I/Q data

So far, we have mentioned sine and cosine as being required to cope with arbitrary phase shifts, but have assumed just a single stream of input samples. The buffer is N samples long but, because of the Nyquist theorem, we can only generate N/2 spectral components, up to half the sampling rate. We are using an in-situ calculation, where the same buffer memory contains input samples and the output spectrum. If it is N samples long, but we only get N/2 spectral components, what lies in the other N/2 memory locations? The answer is a mirror image of the spectrum, a repeat of all the results, but in the opposite direction. The FFT process calculates positive and negative frequencies and a single stream of input samples contains both of these – they are the same – so the spectral output has to show both.

The interesting situation comes if we perform the FFT on I/Q pairs of input samples. We know that I/Q data, depending on the relative phase of the pairs, whether they are 90° or -90° apart, indicates negative or positive frequency. Such an I/Q steam of samples is exactly what we get when an RF signal is down-converted with a quadrature local oscillator in a pair of mixers.

The FFT has 'knowledge' of all negative and positive frequency components present in the input samples, by virtue of their phases, and the resulting spectrum reflects this. Locations 0 to N/2 of the in-situ buffer, after the routine has finished, contain the spectrum of positive frequencies, exactly as it did before, but now the locations from N/2 to N-1 contain the negative frequencies. (Don't be distracted by wondering what a 'negative' frequency is. It is just a part of the complex algebra, and can be displayed on the negative axis of a graph of the spectrum. The frequency is the frequency!) To get a proper continuous spectral display, these upper locations need to be shifted backwards so the last one (location N-1) buts up against location zero. The complete spectrum now runs from $-F_s/2$ through zero to $+F_s/2$. By using quadrature inputs and the full FFT output, we have generated a bandwidth of spectral information equal to the sampling rate. But recall that to do this we must supply I/Q data to start with. So such full-bandwidth spectrum processing is only available for things like digital receivers where the 'negative frequency' is there in the first place, from a quadrature down-converter.

```
  SUB FFT(numsamples&)      'Data in format (I, Q, I, Q…) starting
at index 1'
    nn& = 2 * NumSamples&  'Needed because of using one linear
array for I and Q'
    j& = 1
    FOR i& = 1 TO nn& STEP 2      'Bit reversal first'
      IF j& > i& THEN
        SWAP dat(i&) , dat(j&)
        SWAP dat(i& + 1) , dat(j& + 1)
      END IF
      m& = nn& \ 2          'backslash     is     integer-divide
operation'
      WHILE m& >= 2 AND j& > m&
        j& = j& - m&
        m& = m& \ 2
      WEND
      j& = j& + m&
    NEXT i&
    mmax& = 2

    WHILE nn& > mmax&             'Now do the transform'
      istep& = 2 * mmax&
      theta = 2 * pi / mmax&
      wpr = COS(theta)      'cos  and  sin  value  could  be  pre-
calculated…'
      wpi = SIN(theta)      ' … and stored in a table for speed.'
      wr = 1
      wi = 0

      FOR m& = 1 TO mmax& STEP 2
        FOR i& = m& TO nn& STEP istep&   'DFT butterflies'
          j& = i& + mmax&
          tempr = wr * dat(j&) - wi * dat(j& + 1)
          tempi = wr * dat(j& + 1) + wi * dat(j&)
          dat(j&) = dat(i&) - tempr
          dat(j& + 1) = dat(i& + 1) - tempi
          dat(i&) = dat(i&) + tempr
          dat(i& + 1) = dat(i& + 1) + tempi
        NEXT i&
        wtemp = wr
        wr = wr * wpr - wi * wpi
        wi = wi * wpr + wtemp * wpi
      NEXT m&
      mmax& = istep&
    WEND
  END SUB
```

FIGURE 3: A routine for in-situ radix-2 FFT.

Normal audio input, say from a microphone, can only be processed to a bandwidth of $F_s/2$.

Variations and other types of FFT

Here we have described the radix-2 FFT process, where a buffer length has to be a power of two. This may not always be ideal, and the FFT process can be performed with other radix values. The process is then more complicated to understand, with the reordering particularly so but, where results may only be wanted for a certain range of output frequency data, radix-3, radix-5 or even radix-7 FFTs are possible, or some product of these like radix-6. The buffer length then needs to be a product or power of the radix, which may have certain processing advantages. It is no coincidence that for all the modes within the WSJT-X suite of data modes, all the symbol rates and tone spacings, are exact submultiples of the sampling rate, and these submultiples all have factors of just 2, 3 and 5. FFTs with different radixes form a fundamental part of the decoding process in this software.

The examples given have adopted in-situ calculation where the same buffer is used for time and resulting frequency data. This may not always be convenient, so there is no reason why separate buffers for time-sampled data and calculated frequency bins can't be maintained.

Continuous conversion

The process described above, reading one complete buffer full of input samples, doing the transform, then moving to a completely new set of samples is suitable for the situation where the type of input waveform is static, such as continuous tones and constant-amplitude waveforms that have a constant frequency spectrum. But in reality we want to look at waveforms that change all the time, sometimes quite rapidly, like the RF spectrum, voice waveforms, or the like. If we use the one-buffer-at-a-time process, we are in danger of missing events that change suddenly, especially those that occur close to the boundaries of each block.

To allow maximum opportunity to capture rapid changes, it is usual to perform an FFT on a block, then read in a set of just half a block of new samples, moving the remaining half of the old ones to the end and discarding the other half. Now the FFT process must be performed twice as fast, on overlapping pairs of half-block lengths of data, so each half-buffer of time samples appears in two successive frequency spectra. Occasionally even further overlapping is employed with three, four or even more sub-blocks being employed, with a corresponding increase in the FFT rate.

When overlapping data is worked on in this way, simply reading it into a buffer for in-situ processing is not an option as a portion of that input data needs to be retained for the subsequent transform. The new samples have to be copied into the FFT working buffer from a separate buffer containing all the input samples. This is another application for the circular buffer technique introduced in the digital filtering article. The FFT buffer is then filled starting from the tail pointer of a suitably-long circular buffer.

Appendix 1

Figure 3 shows a routine for in-situ radix-2 FFT written in one dialect of BASIC derived from an original FORTRAN routine. Data is passed into and out of the routine in the global array `dat()` which contains alternate real and imaginary samples, or successive I/Q pairs. The index starts at 1 and continues to 2N for N complex samples. For real-only data, the input is read into alternate 'I' locations with the 'Q' values each set to zero. The bit reversal is carried out first, in a seemingly obscure way in this routine but it works fast enough. Variables ending in '&' are integer values, and those without a suffix are floating-point values.

Design Notes

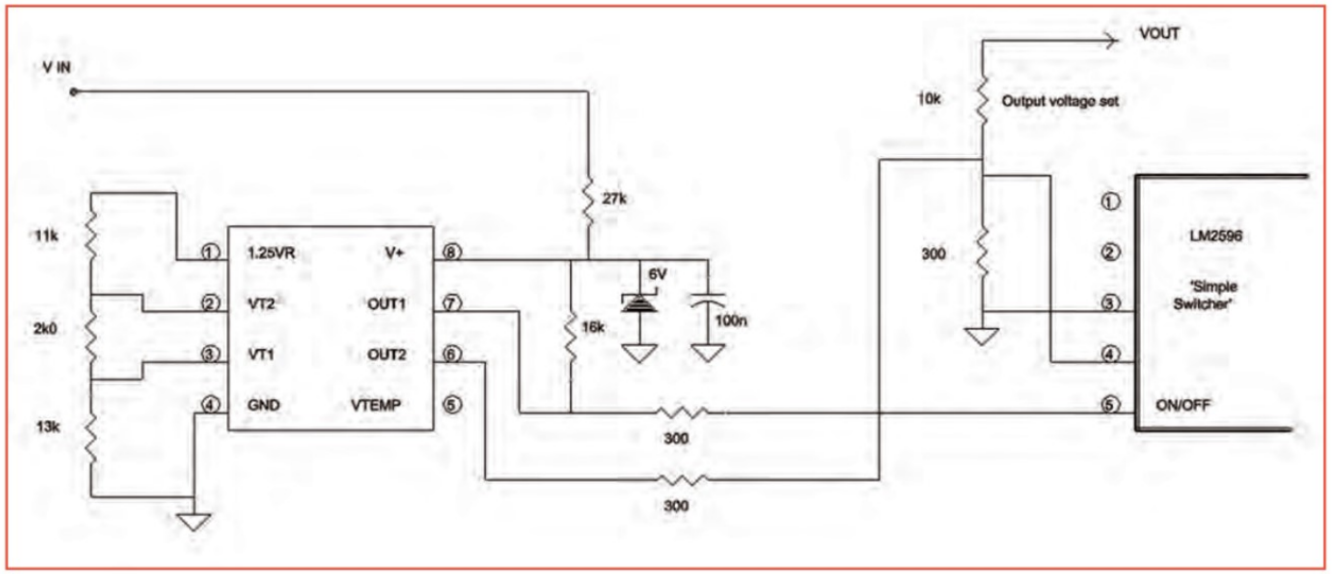


FIGURE 1: The circuit diagram of the two-speed fan control for a solar inverter, using an LM56 temperature-controller chip with an LM2596 'simple switcher'.

BBC Computer history, a correction

It would appear that there was a bit of a tangle in the update of the reporting of the history of the BBC Computer described in March's *Design Notes*, and things didn't go in quite the way described there. Richard Russell, G4BAU, 'the man himself', saw that article and wrote in with these corrections: "My attention has been drawn to a piece in *RadCom* which mentions the Z80 microcomputer I designed whilst working in the BBC Engineering Designs Department. This has rather got its knickers in a twist, because the machine wasn't anything to do with the BBC; it was an entirely private project. It can't in any sense be described as a 'BBC Computer'. The main board was developed jointly by myself and Colin Foddering, hence commonly being referred to as the CRF/RTR board. A companion board, to support bit-mapped (monochrome) graphics, was later developed by myself and Alan Tresadern, known as the AET/RTR board. Its existence did become known within the BBC engineering community, and a few were made, but it never became in any sense an official project. I still have one in the loft, although it hasn't been switched on in many years so I've no idea if it still works." This reminds me of a Motorola MC6809 computer board I built whilst working at the Royal Aircraft Establishment Farnborough in 1982. A friend, at another site, was using a development board for this chip. The PCB had been laid out, and a large batch was built in RAE's own PCB facility, and several departments were using this development module. Designed by a radio amateur, Gordon (whose surname and callsign I don't recall), the PCB was completely self-contained with an LED display of HEX address and memory, a HEX keypad, a simple EEPROM-based operating system, and a cassette-tape interface for long-term storage. There was plenty of address-decoding pads and holes for adding peripheral devices. That's how we used microcontrollers back then! It was called the 'GPC' computer, but whether that stood for 'general-purpose computer' or were Gordon's initials, or both, who knows more than 40 years on? I got one of those PCBs, and built my own 6809 development board, mastering assembly language programming in its absolutely fundamental way for a couple of years before better stuff came along. On showing the hardware to a friend, he said "that's the *Wireless World* Nanocomp," a design I hadn't seen at the time. It wasn't exactly the same design that *Wireless World* had published back then, but was close enough. Gordon had taken that design, evolved his own version, and designed a PCB using facilities available at the time. The original true Nanocomp can be seen at **[1]**, but all traces of the RAE-developed version are long gone.

Fan-controller chip

In the December 2020 column, we looked at a design for controlling a cooling fan, offering proportional or switching control with hysteresis. More recently, quite by accident, I stumbled across a custom fan-controller chip, the LM56. This 8-pin device has an integral temperature sensor, and gives two outputs, each a switch to ground, for controlling a two-speed fan. The two programmable temperature thresholds each have built-in hysteresis, and are set by external resistors. A data sheet for the LM56 can be found at **[2]**.

Back in the hot weather last August, I purchased a solar panel and mini inverter for some solar-power experimentation. When generating at the full capacity of over 300W, the inverter ran excessively warm, becoming almost too hot to touch; it didn't help that is was sitting inside my conservatory heated by strong sunlight. According to the data sheet, the inverter has its own internal protection against overheating, but I wasn't happy; fan cooling was definitely needed. This would only have to come on at temperature extremes, and I was just about to build one to that earlier design when I remembered the LM56. I'd salvaged a couple of these chips

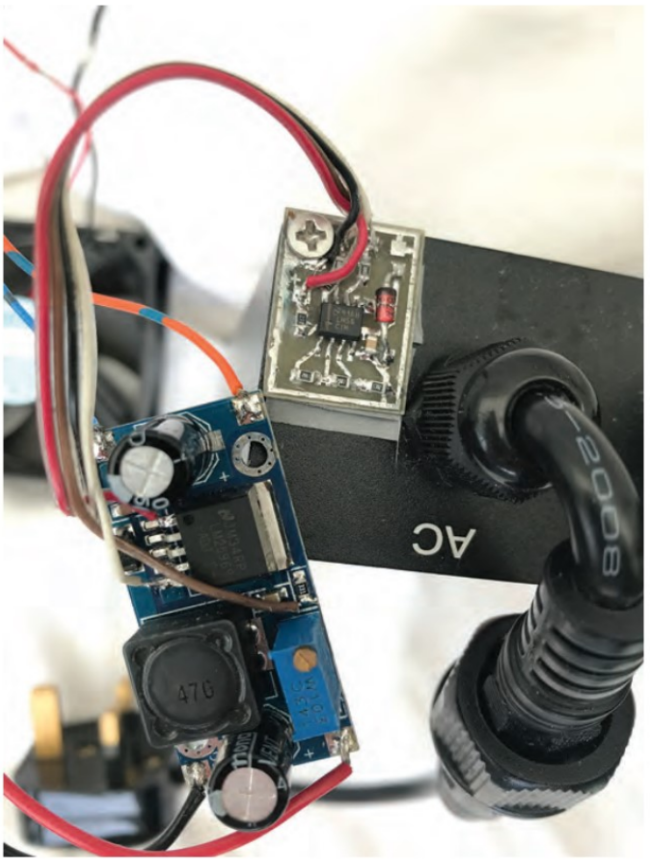


FIGURE 2: **The temperature sensor and power supply modules used for controlling a cooling fan on a solar inverter.**

from old bits of digital hardware and had plenty of fans, both 12V and 24V ones.

The solar panel delivered 35V to 40V to the inverter in bright sunlight, with the exact voltage wobbling around as illumination changed and maximum power-point tracking did its job. To drop the solar panel voltage down to 24V for the fan, I took an off-the-shelf eBay module that used the LM2596 simple switcher chip **[3]**. This device has an enable input on pin 5 that is normally just connected to 0V on the module. It is easy enough to lift this pin to get on-off control, so the switch closure to ground, provided by the LM56, could be connected directly to this pin for one of the temperature thresholds. The other output from the LM56 was used to reduce the output from the switcher for a slower fan speed by connecting it into the voltage-regulation feed-back loop via an appropriately-selected resistor (read that as 'selected by trial-and-error') to set the output voltage. Therefore, with a mild temperature increase, the fan would come on at slow speed, then rise to full speed as everything heated up to uncomfortably-high levels. **Figure 1** shows the circuit I employed, with the LM56 circuit shown in its entirety and just the relevant connections to the LM2596 voltage-setting feedback.

The values chosen for the resistor chain around the LM56 give temperature-switching thresholds at around 37°C and 60°C, and the 1°C of hysteresis that is built into the chip. The data sheet gives full details of the equations for calculating these resistor values for any pair of temperature thresholds. The LM56 device requires a power supply in the range 2.7V to 10V, and takes a miniscule current consumption of 230µA. This was provided by a dropper resistor and a Zener diode from the solar input.

The two threshold outputs switch to ground as temperature rises,

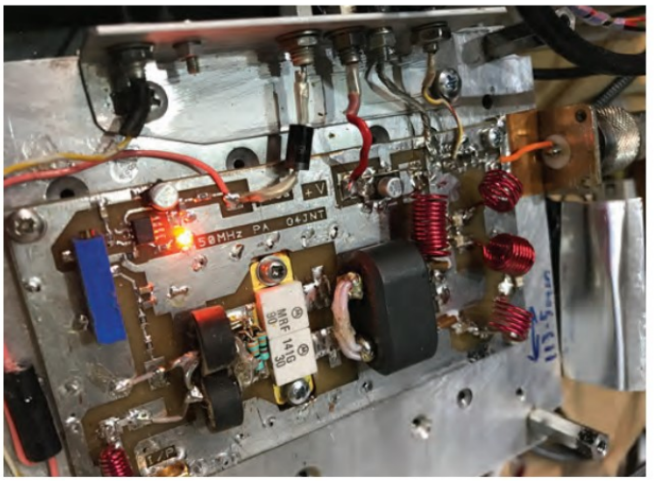


FIGURE 3: **The power amplifier, showing the dual output MOSFET in the GB3MBA 50MHz beacon. This picture was taken after re-flowing solder to repair a fault caused by the top right-hand side drain connection not being properly soldered to its PCB pad.**

which means that, with this minimalistic circuit configuration, the switcher has to be adjusted on its own to the lower of the two desired output voltages, here around 15V. The higher fan voltage of 24V is set when the second threshold output pulls the voltage feedback point lower with a resistor. The values shown in the diagram apply to my LM2596 modules that use a 300Ω resistor in their voltage feedback. Other modules from different suppliers may differ, so some juggling of values may be needed to get the desired pair of output voltages. A photo of the LM56 built onto a small PCB, alongside the modified eBay switcher module can be seen in **Figure 2**. One of these days it all may get mounted properly, but in the meantime the switcher module is happy enough flapping around in free space. In practice, the slow fan speed has usually been enough to cool the inverter, and only on a couple of very-hot sunny days last year, when the ambient temperature itself rose to an uncomfortable level, did the controller switch to maximum speed.

GB3MBA beacon fault

The GB3MBA 50.408MHz beacon is designed for meteor-scatter observation and characterisation, and it has been running happily for about two years. The transmitter is located at the Sherwood Observatory of the Mansfield and Sutton Astronomical Society in Locator IO93JC, and delivers around 80W of carrier to the crossed Moxon antenna. A fast CW identifier is given every ten minutes. Full details of the beacon and how to use it can be found at **[4]**.

Recently, transmissions suddenly died with no power output being reported, although telemetry was still indicating a PA heatsink temperature running only slightly cooler than normal. An antenna feeder fault could be discounted as the RF output power was showing zero, and the power into the dump resistor in the quadrature hybrid on the antenna was also showing zero. The output MOSFET must still have been drawing some current to give this slight warming which was a very worrying situation; such behaviour is typical of one side of the push-pull pair of output FETs having died.

Full details of the PA design can be found in *Design Notes* for

Andy Talbot, G4JNT
andy.g4jnt@gmail.com

November 2021. The power amplifier had been built using a 300W-rated dual MRF141G MOSFET that was therefore being massively under-run. It was cooled by a fan, and used with a power rail slightly lower than its specification, 26V instead of 28V, so there shouldn't have been anything in the design that could have caused failure after just two years of operation. Brian, G4NNS had incorporated this PA module into the complete beacon assembly, adding the PSU, GNSS-locked RF source, and remote internet-based control and telemetry. The failed beacon in its entirety was shipped from its location back to Brian, and I went over to his QTH to investigate the problem.

Probing with a voltmeter

The very first test with the PA powered up was to measure the DC voltage on the drains of the MRF141G device. So, with one of Brian's DVMs set to the 200V range, I placed one lead on the chassis and the other on the drain connections on the two tabs of the PA device. One was at the correct 26V, but the other measured just 19V. Then imagine the (non-electrical) shock that I had when I pressed the probe just a little a bit harder and saw a small spark, accompanied by a small 'crack' sound. This simultaneously crashed the Raspberry-Pi computer used for the beacon controller, removing all power to the PA module. This was a separate issue from the PA failure, and the reason for it is described below.

After reinstating the controller, and trying the voltage probing once again, the same spark/crack happened when the connection was probed, this time with no R-Pi crash. Then Brian suddenly noticed the Bird Thru-line RF-power meter indicated a brief peak in RF output when the probe was pressed on the MRF141G's drain. Examination of the joint under a strong magnifier soon revealed what the problem was; the connection of that drain tab had come adrift, and a tiny gap was just visible between the tab and PCB pad. The other half of the dual-FET had its drain properly connected to the +26V supply, which led to the current draw as the device was being driven with RF, thus forcing that side to conduct some DC. However, as there was no differential RF load across the pair of devices, it couldn't deliver any RF output power. That one-sided current was the reason for the temperature rise. The RF drive to the unpowered half would also have been rectified in the body-diode inherent from source to drain in all MOSFETs, and was no-doubt the reason for the 19V reading that was initially seen on its floating drain connection.

It was clear that, when building the PA, I hadn't flowed enough solder to connect that drain tab properly to the PCB pad, and there must have just been a small whisker, or perhaps even just compression, making the initial connection. Over a couple of years of operation, temperature cycling and perhaps mechanical jolting, that whisker had decided to break causing the failure. My probing it with the voltmeter closed the fraction-of-a-millimetre gap and the PA fired-up.

After some solder reflowing to reconnect the tab, and a bit more solder to one of the gates that also looked as if it might benefit from it, the PA was working fine again. Brian then reintegrated the PA into the rest of the beacon and shipped it back to its final location. At the time of writing, the web page appears to show the beacon has just been reinstalled and is working correctly.

Mounting RF power devices

The problem came about because of the way RF power devices need to be mounted. The source connection is often just the mounting flange, with no other tabs on it to solder. The standard way of mounting these on a heat sink is to mill a gap so that, when the device is placed in this gap, its drain- and gate-mounting taps sit just high enough above the adjacent un-milled surface to accommodate the thickness of the PCB. A small amount of solder then fills the residual gap. The ground plane on the underside of the PCB lies flat on the heat sink. Clamping this securely, or using low-melting-point solder to bond the ground plane to heat sink, ensures the very-low RF impedance essential between device source and the PCB ground plane.

Here at 'JNT labs, in the absence of a milling machine, I don't do it this the 'proper' way. Instead I use a technique that probably provides a better RF grounding connection. I mount the device directly onto the flat heat-sink surface, then fix the PCB using a couple of washers as spacers under its mounting screws. This raises the PCB above the heat sink just enough for the tabs to sit on their pads. To give the low source impedance from ground plane to the flange, I then run a piece of soft copper foil right across the ground plane where it is cut out for the MOSFET, forming it around the device's mounting flange. The FET is then bolted down with this copper foil between its flange and the heat sink. Being soft copper, it easily forms around the flange and does not degrade the thermal resistance significantly. It may actually help, as soft copper flows and is good at filling minute gaps. And, of course, being continuous and soldered to the PCB ground plane in this solution, it gives a shorter RF path than the traditional mounting method can provide.

However, in this case, the washers used as spacers for PCB-to-heat-sink spacing had resulted in a gap of something like 0.5mm between the FET tabs and PCB pads. The tabs should never be forced hard to bend them, as this could break the seal on the ceramic package on the MRF141G. As the tabs are very short, I wasn't going to try to bend them down much to mate with the PCB. The gap needed to be filled with solder, and I clearly hadn't used enough to get a proper flow underneath the tab to wet both surfaces properly. The one whisker that did connect gave a false sense of security when the amplifier passed its functional test, and only decided to give up after two years. **Figure 3** shows a photograph of the PA device after the solder had been re-flowed to make a proper robust connection.

Parallel grounding paths

The incidental issue, that crashed things during the first stage of the beacon fault-finding, is something that needs to be remembered when building equipment that consists of several modules bolted onto metal racks or chassis. The first DVM probing, that created the initial spark, crashed the R-Pi computer and deactivated the whole beacon controller. This turned out to be a simple grounding issue. When dismantling the beacon to remove the PA module and place it on the bench under test, the ground connection, that should have been in place by virtue of the PA being screwed to a metal base plate, had gone because two sections of chassis had been separated. The DC return current for the PA was only being carried by the outer of the thin co-ax and an SMC connector delivering the RF input from the GNSS locked source. The sharp pulse of current in this high impedance ground loop, as the PA sharply kicked into life (the spark), was enough to crash the computer, fortunately non-catastrophically. After reinstating a good bond between these two bits of the chassis, the bench testing could proceed as described above.

So the lesson to be learned here is: do not rely on a module being bolted to a chassis to give a return DC path. Always include a parallel wire link. Also include one between different mounting plates, just in case. Then, when the module is removed for testing, or the unit is split but still needs to remain connected, nothing changes in the grounding, so forcing current to go along paths it shouldn't go along.

References

[1] Wireless World Nanocomp: https://www.hackster.io/news/reproducing-a-retro-8-bit-computer-design-on-a-breadboard-219db2cc7127
[2] LM56 Datasheet: https://www.ti.com/lit/ds/symlink/lm56.pdf
[3] LM2596 simple switcher chip: https://www.ti.com/lit/ds/symlink/lm2596.pdf
[4] GB3MBA meteor beacon: https://ukmeteorbeacon.org/Home

Review

Kenwood TH-D75

As I've observed before, handheld transceivers seem to fall into two very-diverse categories: 'value' (I recently paid under £10 for a very serviceable and basic handheld), and 'premium', where there are many additional facilities on top of the basic 2m/70cm FM transceiver capabilities.

FIGURE 1: The TH-D75 is a solid unit and is well built.

An expensive handheld transceiver

The Kenwood TH-D75 (see **Figure 1**) is firmly in the 'premium' category, with a £790 price tag to match, making it, I think, the most-expensive handheld currently in the amateur market. What does the TH-D75 do on top of basic operation, and does it give good value for money? Well, highlights of the rig include dual-watch digital voice operation, a wide-band HF receiver, analogue and digital automatic packet reporting system (APRS), integrated packet digipeater as well as USB-C charging, and the ability to connect a computer to gain access to the rig's internal packet terminal node controller.

Let's see how it worked out in practice. Unpacking the rig, it feels a nice solid unit and is well built. Switching on brings the colour screen to life. The screen is a good size and is easy to read. The rig reminded me strongly of its predecessor, the TH-D74. Not only is the TH-D75 an analogue (FM) radio, but it 'does' D-Star as well. The Kenwood TH-D74 was the first non-Icom rig to have D-Star built in, and the TH-D75 continues this tradition.

Usability

Out of the box, I found the basics of the TH-D75 fairly easy to master, without my having to study the manual, although some of the key combinations didn't seem entirely intuitive to me. I remember thinking the same about the TH-D74, so I suppose if you are used to that rig and are upgrading, you will find it straightforward. If you are used to handhelds from other manufacturers, however, then you may experience, as I did, a bit of a learning curve. Of course, as you become used to the rig and use it frequently, it will become second nature. The supplied user manual is fairly superficial, so for any detail you will need the full manual **[1]**.

Getting started on D-Star

My first test, then, was to use the TH-D75 with my MMDVM hotspot to connect to one of the D-STAR networks. I set my callsign in the radio, switched to DV mode, set the frequency and was delighted that I could hear D-Star traffic from the hotspot, and of good audio quality too. The fun started when I tried to respond to stations on the hotspot. For some reason, the hotspot wasn't recognising the transmissions from the TH-D75. I tried setting a repeater shift of + 0Hz (a trick that works with the Icom radios), but no, that didn't work. I had a feeling that I probably needed to be in digital repeater (DR) mode rather than DV, and more specifically, to be able to set a gateway callsign, eg 'GW4VXE G'. The problem was that I couldn't find a way of programming that from the front panel. Actually, as it turns out, I could have used the route select function from the menu, but I didn't realise that at the time! It was interesting that I needed to configure this manually. In Icom's D-Star implementation, the rig automatically notes the gateway setting and updates it, so you don't have to make this configuration step.

The TH-D75 comes configured with a large number of D-Star repeaters which you can use in DR mode. I believe this list can be refreshed as repeaters come and go. This is all good but, of course, my hotspot wasn't in the list.

So, I installed the USB driver for the TH-D75 together with the Kenwood memory programming software (a free download) onto my Windows PC, and then connected the TH-D75 using a USB-C cable (not supplied), and added an entry into the repeater list for my hotspot including, critically, the RPT2 entry 'GW4VXE G'. I uploaded that back to the TH-D75 and was immediately able to use my hotspot fully. Although I had set up my hotspot as a repeater in the repeater list, there was also the ability to save up to 30 hotspots in the TH-D75, and to select them as required.

The TH-D75 contains a GPS receiver, so you can opt to send your position with your D-Star transmissions so that, when someone receives your transmissions (assuming that their rig also has GPS), they will be able to see a range and bearing to your position. This can be quite interesting. Equally, on receive, if the TH-D75's GPS is switched on and it has found sufficient satellites to establish its position, it will show you range and bearing to any GPS-equipped D-Star signals that you receive.

The dual-watch facility means that you can, if the occasion arises, monitor two D-Star channels at the same time.

Detailed D-Star instructions are not shown in the supplied user manual, but can be found in the full manual.

FIGURE 2: **The TH-D75 in use with a digital voice mode.**

FM operation

My first test was to try out the receiver on 70cm, with the EI7MLR repeater across the Irish Sea, some 85mi away, and the TH-D75 worked as expected. Setting the repeater shift, CTCSS tone, and so on, was all quite straightforward. Similarly, setting up the TH-D75 to use my Allstar node (analogue hotspot) was straightforward, and the audio quality on both transmit and receive was very good. For use with hotspots, the TH-D75 is equipped with an 'economic low-power mode' of 0.05W, which is ideal, both from the perspective of battery life, and also if there are any other hotspot users nearby on the same frequency.

Selectable power levels with the TH-D75 are 5W, 2W, 0.5W and 0.05W, sensible power choices for most people. The battery supplied has a capacity of 1820mAh, which seemed to last well while the radio was on test here. A charger is supplied (interchangeable with previous Kenwood chargers), and charging the battery to full capacity took a couple of hours using this method. USB-C charging is also available which can be useful, but you'll need to supply your own USB-C cable.

The TH-D75 does not operate in full-duplex mode, unlike the TH-D72, which is a shame for satellite operators. What *is* good for satellite operators, however, is the ability to receive (only) SSB and CW signals on 2m and 70cm, so you could easily use the TH-D75 as the receiver in your satellite station, and transmit SSB or CW on another rig.

HF reception

The TH-D75 features a receiver capable of receiving in the following frequency ranges: 0.3MHz to 76MHz, 118MHz to 174MHz, 200MHz to 250MHz (including AM), 382MHz to 412MHz and 415MHz to 524MHz. For the HF ranges, you have the choice (switchable from the menu) of choosing between the antenna connected to the SMC-F socket, or an internal ferrite-bar antenna. At least, in the shack, the ferrite-bar antenna didn't do much more than pick up noise and some weak signals. However, I connected a cheap 27MHz portable antenna to the SMC socket and, as you might hope, on 10m, signals were really excellent. The 28.074MHz FT8 frequency was well over S9, and plenty of CW and SSB signals were easy to hear. I tried several other bands, including 18MHz and 21MHz, and the receiver was very good indeed. You can tune in 20Hz, 100Hz, 500Hz or 1000Hz steps. Whereas I sometimes think that HF reception on a handheld is a bit of a gimmick that you probably wouldn't use, theTH-D75 is so very good that I regard it as a real plus point.

Marine and air band reception

The TH-D75 receives well on the marine band, and civil and military air bands.

APRS capability

Like the THD-74 and the THD-72 before it, the THD-75 has a very good APRS implementation. You can, for example, take the TH-D75 out for a walk, and have it send your position as a beacon which can then be picked up by other APRS stations. If you have it working on receive, it will display data from the APRS packets that it receives, showing what sort of station it was and a distance and bearing to that station (assuming that both stations are running). You can send APRS messages (like text messages) to other APRS-equipped stations. You can even send APRS messages through the digipeater on the International Space Station, but you'll need to change the packet path to do this, and you'll probably want a better antenna than the supplied 'rubber duck' antenna, although that will work when things are quiet. Unfortunately, I wasn't able to test the APRS functionality on the TH-D75 fully, as my local APRS digipeater was off the air because of bad weather during the review period.

APRS is the most commonly-used implementation of packet radio these days. However, if you are one of those enjoying revisiting packet radio, you'll be pleased to

Tim Kirby, GW4VXE
gw4vxe@icloud.com

know that the THD-75 includes a terminal node controller (TNC), but a word of caution: although it supports 1200- and 9600-baud operation, the TNC only operates in the 'keep it simple, stupid' (KISS) mode, rather than having command or converse modes, so you'll need software on your computer which implements KISS mode, rather than a simple terminal program.

Bluetooth

The THD-75 comes with Bluetooth built in, so you can pair it with a Bluetooth headset.

Micro SD and recording

The TH-D75 contains a slot for a micro SD card, which can be used for recording audio files. You can set the TH-D75 to record any traffic that it hears on either band A or B. Recordings will be saved in waveform audio-file format (WAV). I am never entirely convinced how useful this feature is, although most manufacturers seem to include it, so I guess there must be a demand. It could be useful for monitoring unusual (or unwanted) traffic, perhaps.

FM radio

The TH-D75 comes with an FM radio. This is another facility that seems to come with almost every handheld, but how many of us actually use it? It's there if you want it!

Overall impressions

My first impression of the TH-D75 was not particularly enthusiastic; the menus seemed harder to navigate than I had expected, and I had some problems getting D-Star going through my hotspot. However, these frustrations dissipated as I spent more time with the rig, and I enjoyed using it. I particularly liked the multimode HF/VHF/UHF receiver, which could be very enjoyable to use if you are out and about relaxing on a beach, hilltop, etc. The D-Star implementation is quite attractive; it looks good and works well, though perhaps there is a little more configuration to do than with the Icom D-Star radios. On FM, the rig worked well, and it was easy to save memory channels, configure analogue repeaters, and so on. It's worth saying that the rig doesn't come loaded with a directory of analogue repeaters, only D-Star ones.

Who will the TH-D75 appeal to? To get the best out of the rig, you'll need to be a keen D-Star user, I suspect, perhaps with a hotspot at home and an enjoy of D-Star (and analogue repeaters) when out and about. The APRS implementation is good, and if you are interested in packet radio in a wider sense, then the built in TNC and digipeater will be useful and of interest to you. The natural competition for the TH-D75 will be the Icom ID-52 (£550). The TH-D75 has the multimode HF/VHF/UHF receiver, and a wider packet implementation, whereas the ID-52 has a better repeater directory (analogue as well as digital), some band-scope capability (not present on the TH-D75), as well as some ancillaries like picture exchange, but many of the other features are pretty much like for like. If you are in the market for a top-of-the-range D-Star-capable hand-held radio, you'll want to decide which of these are important to you. I hope I've given you the information you need to do that.

The TH-D75 is available for £790 (inc VAT). My grateful thanks to Martin Lynch and Sons for the loan of the unit.

Reference

[1] http://manual.kenwood.com/files/B5A-4505-00_01_EN.pdf

Contest Calendar May 2024

Ian Pawson, G0FCT

RSGB HF Events

Date	Event	Times (UTC)	Mode(s)	Band(s)	Exchange
Mon 13 May	80m Club Championship	1900-2030	SSB	3.5	RS + SN
Mon 20 May	FT4 Series	1900-2030	FT4	3.5-28	Report
Wed 22 May	80m Club Championship	1900-2030	DATA	3.5	RST + SN
Thu 30 May	80m Club Championship	1900-2030	CW	3.5	RST + SN

RSGB VHF Events

Date	Event	Times (UTC)	Mode(s)	Band(s)	Exchange
Wed 1 May	144MHz FT8 AC (4 hour)	1700-2100	FT8	144	Report + 4-character Locator
Wed 1 May	144MHz FT8 AC (2 hour)	1900-2100	FT8	144	Report + 4-character Locator
Sat 4 May	432MHz Trophy	1400-2000	All	432	RS(T) + SN + Locator
Sat 4-Sun 5 May	May 432MHz – 245GHz	1400-1400	All	432-245G	RS(T) + SN + Locator
Sun 5 May	10GHz Trophy	0800-1400	All	10G	RS(T) + SN + Locator
Tue 7 May	144MHz FMAC	1800-1855	FM	144	RS + SN + Locator
Tue 7 May	144MHz UKAC	1900-2130	All	144	RS(T) + SN + Locator
Wed 8 May	432MHz FT8 AC (4 hour)	1700-2100	FT8	432	Report + 4-character Locator
Wed 8 May	432MHz FT8 AC (2 hour)	1900-2100	FT8	432	Report + 4-character Locator
Thu 9 May	50MHz UKAC	1900-2130	All	50	RS(T) + SN + Locator
Sun 12 May	70MHz CW	0900-1200	CW	70	RST + SN + Locator (UK also sent Postcode)
Tue 14 May	432MHz FMAC	1800-1855	FM	432	RS + SN + Locator
Tue 14 May	432MHz UKAC	1900-2130	All	432	RS(T) + SN + Locator
Thu 16 May	70MHz UKAC	1900-2130	All	70	RS(T) + SN + Locator
Sat 18-Sun 19 May	144MHz May	1400-1400	All	144	RS(T) + SN + Locator (UK also send Postcode)
Sun 19 May	1st 144MHz Backpackers	1100-1500	All	144	RS(T) + SN + Locator (UK also send Postcode)
Tue 21 May	1.3GHz UKAC	1900-2130	All	1.3G	RS(T) + SN + Locator
Tue 28 May	SHF UKAC	1830-2130	All	2.3G-10G	RS(T) + SN + Locator

Best of the Rest Events

Date	Event	Times (UTC)	Mode(s)	Band(s)	Exchange (Info)
Sat 4 May-Sun 4 Aug	UKSMG Summer Marathon	0000-2359	All	50	4-character Locator
Sat 4-Sun 5 May	ARI International DX	1200-1159	CW, RTTY, SSB	3.5-28	RS(T) + SN (I give Province code)
Sun 5 May	UKuG Low Band	0800-1400	All	1.3-3.4G	RS(T) + SN + Locator
Sun 5 May	UKuG mm-wave	0900-1700	All	24G, 47G, 76G	RS(T) + SN + Locator
Sun 5 May	WAB 7MHz Phone	1000-1400	AM, FM, SSB	7	RST + SN + WAB
Sat 25-Sun 26 May	CQWW WPX CW	0000-2359	CW	1.8-28	RST + SN
Sun 26 May	UKuG High Band	0600-1800	All	5.7G, 10G	RS(T) + SN + Locator

For all the latest RSGB contest information and results, visit www.rsgbcc.org

Feature

Trans-Equatorial propagation

Trans-Equatorial Propagation (TEP) is a fascinating phenomenon that allows for the propagation of VHF and UHF signals over long distances, particularly along a north-south path that crosses the magnetic equator. While TEP has been observed and studied for many years, the exact mechanisms behind it are complex and not fully understood.

The science and mathematics behind TEP

The ionosphere, a layer of the Earth's atmosphere that is ionised by solar and cosmic radiation, plays a crucial role in TEP. The ionosphere is often modelled as a series of horizontal layers that vary with time, location, and sunspot activity. However, the real ionosphere, particularly in equatorial and polar regions, is much more complex.

One of the key features of the equatorial ionosphere that gives rise to TEP is the equatorial anomaly. This is where a high electron concentration is found on each side of the magnetic equator, usually seen in the region of 10 to 20° latitude. The afternoon TEP is believed to occur when a signal is reflected first by an anomaly on one side of the equator and then again by another anomaly on the other side.

Evening TEP is less well understood but is believed to rely on 'ionospheric bubbles' – areas of high ionisation density off which signals are reflected. Other features of the ionosphere that give rise to these unusual modes include sporadic-E, the equatorial ionisation enhancements, ionospheric tilts at twilight, and ionospheric irregularities such as equatorial spread-F.

The mathematical modelling of TEP involves understanding the behaviour of radio waves as they interact with these ionospheric layers. This is typically done using the principles of electromagnetic wave propagation, which are governed by Maxwell's equations. The propagation of signals in TEP is often modelled using ray-tracing techniques, which involve solving a set of differential equations that describe the path of the radio wave as it travels through the ionosphere. These equations consider the varying electron density in the ionosphere, as well as the frequency of the radio wave.

Web-based radio systems studying TEP

With the advancement of technology, web-based radio systems have become a valuable tool in studying phenomena like TEP. These systems allow for real-time monitoring and data collection from various locations around the world, providing a wealth of information for researchers. They can track changes in signal strength, propagation times, and frequencies of occurrence, contributing to our understanding of TEP **[1]**.

Recent breakthroughs in TEP research

There have been several recent breakthroughs in TEP research. A study by Keisuke Hosokawa and his team investigated the feasibility of monitoring equatorial plasma bubbles (EPBs) using VHF radio waves used for aeronautical navigation systems. This study represents a significant step forward in the use of existing infrastructure for the wide-area monitoring of EPBs **[2]**.

A comprehensive overview of TEP was provided, detailing its historical context, occurrence times, and the characteristics of afternoon and evening TEP. This resource serves as a valuable reference for both newcomers and experienced researchers in the field **[3]**.

How can radio amateurs contribute?

Radio amateurs can contribute to the study and understanding of TEP in several ways. By operating on the VHF and UHF bands, particularly around the equinoxes when TEP is most prevalent, radio amateurs can collect valuable data on signal strength, propagation times, and frequencies of occurrence. Sharing these observations with the scientific community can provide real-world data to support theoretical models and predictions **[4]**.

Amateurs can also conduct their own experiments to test theories and hypotheses about TEP. This could involve varying the frequency, time of day, or antenna configuration to see how these factors influence TEP influence TEP by utilising automated reporting tools and then analysing the results.

Practical examples of TEP observations

TEP was first noticed in the 1940s by both military and amateur operators who discovered that it is possible to communicate in the VHF band over intercontinental distances during times of high sunspot activity. The first organised, and therefore relatively large scale, TEP communications occurred during 1957-1958 in the peak of sunspot cycle 19 **[5]**.

There are two distinctly different types of TEP that could occur. The first type occurs during the late afternoon and early evening hours and is generally limited to distances under 6000km. Signals propagated by this mode are limited to the low VHF band (<60MHz), are of high signal strength and suffer moderate distortion (due to multipath). The second type occurs around 8pm to 11pm local time and is more frequent around the equinoxes and especially at times of high sunspot activity. Signals may have doppler spread, are subject to rapid fading and strong distortion, and path lengths are usually between 3000 and 8000 kilometers.

TEP can also occur in the late morning hours, allowing for radio communication between southern UK and Southern Europe (Greece, Malta, Spain) and South Africa. During these times, the ionospheric conditions can align in such a way that VHF signals are able to travel over the equator, allowing for communication between these regions. In some cases, as per personal experiences and observations, these distances can extend beyond 9000km (ZS6 - OH7) and (ZS6 - OY).

While VHF TEP is more commonly discussed, UHF TEP also occurs, albeit less frequently. In the article *Transequatorial Propagation, TEP: Everything You Need to Know*, it is mentioned that workable contacts have been made on 144MHz (2m band) and sometimes on 432MHz (70cm band), which falls in the UHF range of frequencies. This shows that while UHF TEP is less common than VHF TEP, it is indeed possible and has been observed by radio operators.

Ongoing research

The International Telecommunication Union (ITU) is actively involved in ongoing research on TEP. They have published a recommendation on the method for calculating sporadic-E field strength, which is relevant to TEP. The ITU Journal has also called for papers on propagation modelling for advanced future radio systems, which includes TEP. Furthermore, a study on monitoring equatorial plasma bubbles using aeronautical navigation systems is a recent development in this field [2].

A radio amateur can contribute a great deal into the study of the propagation and its characteristics, helping to provide essential detail on observations. The first step in the analysis process is always to collect sufficient data and the radio amateur community is well placed to make such a sizable contribution, either through the PropNet or BeaconSpot networks.

Observations are quite easy to make once there is an idea of what and where signals are. The amateur radio community benefits from fixed frequencies for computer-generated modulation schemes, generically called MGM and currently consists of either FT8 or FT4 modes. Contacts between radio amateurs can be automatically logged to the WSPRNet servers on 28MHz, 40MHz, and 50MHz **[6]**. In addition, there are a set of highly-reliable transmitters beaconing their

callsign details continuously so that, provided someone is listening, if a propagation path occurs there is a great chance the occurrence will be logged.

The 40MHz band has a small number of beacons running and these are very good indications for possible TEP and other anomalous propagation modes. This band provides an excellent bridge into the low VHF region.

The beacons can be found in **Table 1**.

40MHz beacons

From this location in south-east England, ZS6WAB is currently received (early April 2024) at good strength for about 15 to 20 minutes in the late morning (between 1030 to 11.30UTC). Shortly afterwards the signal returns considerably weaker before fading out completely.

The characteristic observation is that the signal at 40MHz suddenly appears increasing in strength before fading out after 30 to 45 minutes. Then, an hour or so later, a much weaker signal appears with the same fast onset and slow decay profile before fading out completely for the day. This phenomenon seems to have been also observed by other observing stations.

The signal itself is usually T9, ie shows no sign of dispersion (no Doppler spread). Signal amplitude varies but can usually be classed as medium fading.

A closer Look at ZS6WAB on 40MHz

For some reason yet to be determined, parallel reception of ZS6WAB on 28MHz and 50MHz has yet to be noted at this reception location.

At present, there is limited data available so that the current observation window is restricted from late March to the first week of April 2024. This is best illustrated as shown in **Table 2**.

A general trend showing the expected late morning (UK), early afternoon (ZS) propagation is clear. Observing stations have either horizontal or vertical polarisation, suggest 'simple' antennas in use and reported signal strengths appears to be very similar from southern to northern European observers.

Signals also appear to be described as having 'T9' with some fading.

Conclusion

The study of TEP is a fascinating area of research that continues to challenge and engage the scientific community. While we have made significant strides in our understanding of this phenomenon, there is still much to learn. Through continued research and collaboration between scientists and radio amateurs, we can continue to unravel the mysteries of TEP and enhance our understanding of this complex and intriguing phenomenon. The ongoing research in this field is testament to the complexity and intrigue of TEP. It's a fascinating area of study that continues to challenge and engage the scientific community. The practical examples and observations of TEP provide valuable insights into this phenomenon and contribute to our understanding of this fascinating area of science. As radio amateurs, we have a unique opportunity to contribute to this field and help advance our understanding of the world around us.

Let's continue to explore, observe, and share our findings with the world.

Nicholas Shaxted, G4OGI
nick@g4ogi.uk

References

[1] https://www.amateur-radio-wiki.net/trans-equatorial-propagation/
[2] Hosokawa, K., Saito, S., Nakata, H., Lin, C. H., Lin, J. T., Supnithi, P., Tomizawa, I., Sakai, J., Takahashi, T., Tsugawa, T., Nishioka, M., & Ishii, M. (2023). Monitoring of equatorial plasma bubbles using aeronautical navigation systems: a feasibility study.
[3] https://www.uksmg.org/content/Investigation.htm
[4] ITU Report - ITU-R P.2291-1 (05/2019) and ITU Report - ITU-R P.2345- 0 (08/2015)
[5] Transequatorial Propagation https://www.sws.bom.gov.au/Category/Educational/Other%20Topics/Radio%20Communication/Transequatorial.pdf
[6] https://www.wsprnet.org

Table 1: 40MHz beacons.

Beacon	Frequency	Locator	MGM	Last QRG	ODX (km)	Status
ZS6WAB	40.675	KG46rb	FT8	40.6748	12469	On
GB3MCB	40.050	IO70oj	FT8	40.0500	6781	On
EI1KNH	40.013	IO63ve	FT8	40.0130	4539	On
OZ7IGY	40.0702	JO55wn	PI4	40.0710	9479	On
S55ZMS	40.670	JN86cr	PI4	40.6750	8378	On
EI1CAH	40.016	IO53ck	PI4	40.0160	7585	On
ZS6OB	40.680	KG44de	A1A	40.6800	0	On
WM2XCS	40.685	FN20wv	A1A	40.6850	7106	On
WM2XCW	40.680	CN88lx	–	40.6800	2941	unkn
ZL2WHO	40.687	RE79tp	–	40.6870	0	Off

Table 2: 40MHz reception of ZS6WAB March to early April 2024. Source: Beaconspot.uk

Date	Time	Beacon	Frequency	RPT	Dist.(km)	Spotter
05/04/2024	12:13	ZS6WAB	40.6748	539	8187	F4CXO
05/04/2024	10:58	ZS6WAB	40.6748	519	8187	F4CXO
05/04/2024	10:59	ZS6WAB	40.6750	519	8303	F6ACU
05/04/2024	11:32	ZS6WAB	40.6747	569	8836	G0API
05/04/2024	11:20	ZS6WAB	40.6748	549	8783	G4OGI
04/04/2024	12:05	ZS6WAB	40.6748	529	7824	9A5CW
04/04/2024	11:35	ZS6WAB	40.6748	539	8187	F4CXO
04/04/2024	11:21	ZS6WAB	40.6750	419	8303	F6ACU
04/04/2024	11:39	ZS6WAB	40.6748	559	8784	G4OGI
03/04/2024	11:09	ZS6WAB	40.6748	539	8187	F4CXO
03/04/2024	13:11	ZS6WAB	40.6750	529	7485	IK0OKY
03/04/2024	11:23	ZS6WAB	40.6750	51	9341	OH7PS
03/04/2024	13:07	ZS6WAB	40.6746	559	7841	S59GS
02/04/2024	10:43	ZS6WAB	40.6750	51	7824	9A5CW
02/04/2024	13:18	ZS6WAB	40.6750	559	7819	EA3ERE
02/04/2024	10:50	ZS6WAB	40.6748	529	8187	F4CXO
02/04/2024	10:51	ZS6WAB	40.6748	559	8249	F4FRQ
02/04/2024	11:08	ZS6WAB	40.6750	539	8303	F6ACU
02/04/2024	10:39	ZS6WAB	40.6748	549	8783	G4OGI
02/04/2024	13:15	ZS6WAB	40.6746	559	7841	S59GS
01/04/2024	11:44	ZS6WAB	40.6750		8622	DH6JL
01/04/2024	11:54	ZS6WAB	40.6748	579	8187	F4CXO
01/04/2024	10:26	ZS6WAB	40.6748	519	8187	F4CXO
01/04/2024	11:33	ZS6WAB	40.6750	579	8303	F6ACU
01/04/2024	11:30	ZS6WAB	40.6748	549	8784	G4OGI
01/04/2024	12:33	ZS6WAB	40.6750	559	7485	IK0OKY
01/04/2024	12:06	ZS6WAB	40.6750	529	7485	IK0OKY
31/03/2024	11:28	ZS6WAB	40.6750	559	7819	EA3ERE
31/03/2024	10:40	ZS6WAB	40.6748	539	8187	F4CXO
31/03/2024	11:45	ZS6WAB	40.6750	549	8303	F6ACU
31/03/2024	11:14	ZS6WAB	40.6745	579	7841	S59GS
30/03/2024	10:47	ZS6WAB	40.6750	559	7819	EA3ERE
30/03/2024	10:21	ZS6WAB	40.6750	529		EI7HBB
30/03/2024	10:27	ZS6WAB	40.6748	559	8187	F4CXO
30/03/2024	09:38	ZS6WAB	40.6748	539	8187	F4CXO
30/03/2024	09:31	ZS6WAB	40.6748	519	8187	F4CXO
30/03/2024	10:25	ZS6WAB	40.6750	539	8303	F6ACU
30/03/2024	10:30	ZS6WAB	40.6747	559	8836	G0API
30/03/2024	10:46	ZS6WAB	40.6750	599	8647	OR7T
29/03/2024	09:15	ZS6WAB	40.6750	539	7819	EA3ERE
29/03/2024	12:15	ZS6WAB	40.6750	559	9167	EI2IP
29/03/2024	11:32	ZS6WAB	40.6748	549	8187	F4CXO
29/03/2024	10:32	ZS6WAB	40.6750	519	8303	F6ACU
29/03/2024	11:05	ZS6WAB	40.6748	559	8836	G0API
28/03/2024	11:44	ZS6WAB	40.6750	559	7819	EA3ERE
28/03/2024	11:11	ZS6WAB	40.6748	529	8187	F4CXO
28/03/2024	12:26	ZS6WAB	40.6748	539	8249	F4FRQ
28/03/2024	12:08	ZS6WAB	40.6750	219	8303	F6ACU
28/03/2024	11:19	ZS6WAB	40.6748	529	8836	G0API
27/03/2024	12:33	ZS6WAB	40.6750	559	7819	EA3ERE
27/03/2024	10:39	ZS6WAB	40.6748	579	8836	G0API

Computer control of budget antenna rotators

Having the ability to control an antenna rotator remotely enables lots of possibilities.

However, the cost barrier for computer control can be prohibitive. In this article, I will describe my implementation (**Figure 1**) which enables full computer control using a relatively-inexpensive rotator available from most amateur radio stores, and an infra-red (IR) USB dongle.

Introduction

I am using a Sharman Multicom AR-600XL VHF/UHF Antenna Rotator (**Figure 2**), and the newer generation Flirc USB infrared dongle (**Figure 3**).

A block diagram of my implementation is shown in **Figure 4.** From right to left:

- the rotator is mounted on the mast which is powered and controlled via a three-core cable from a controller box within the shack; the controller box comes with an infrared remote-control;
- I have replaced the remote control with an infrared dongle and custom software that sends signals to the rotator controller from a computer; and
- amateur radio software such as the optional Web GUI (that supports rotator control), can be configured to use this custom software either directly on the same computer or via your Wi-Fi and thereby control the rotator.

In order to duplicate my implementation, you will need to be reasonably proficient with computer command-line operation, although you should be able to get by with limited knowledge. The system will work on Windows, macOS or Linux. In what follows, I give an outline of what needs to be done, and so this is not a full command-by-command explanation. Please get in touch with me by email if you need more detail.

Most software that supports rotator control will interface with HamLib 'rotctl'. HamLib is a rich set of libraries for all things to do with computers and amateur radio. I decided to make a python3 implementation that mimics HamLib, so that any software that controls rotators, and that supports HamLib rotctl, should be able to use my implementation. I doubt that it would be appropriate for me to extend HamLib, as I suspect the other rotator implementations may be more robust, but if you are a maintainer of HamLib, please let me know if you're interested.

In this article I will lay out the capabilities of the Sharman rotator, look at the infrared controller and other options (including building your own for pennies), go through my 'rotctlpy' implementation including using the scripts and 'webgui', and finally show you how to integrate it all with GPredict, a real-time satellite-tracking and orbit-prediction application, that can control the rotator automatically tracking satellites across the sky.

FIGURE 1: The antenna rotator mounted on a pole fixed to my roof.

FIGURE 2: The Sharman Multicom AR-600XL antenna rotator.

FIGURE 3: The Flirc USB dongle.

The Sharman rotator

The AR-600 has a programmable antenna controller with IR remote-control (Figure 2). It can store up to twelve antenna directions, and covers the full 360° in azimuth. It is well-suited for mounting on a permanent outdoor mast, and has an approximate rotation time of seventy-four seconds. Unfortunately, the rotator does not support an elevation drive, so the elevation is therefore fixed. Once

Steven Dodd, M0SNZ
stevendodd@outlook.com

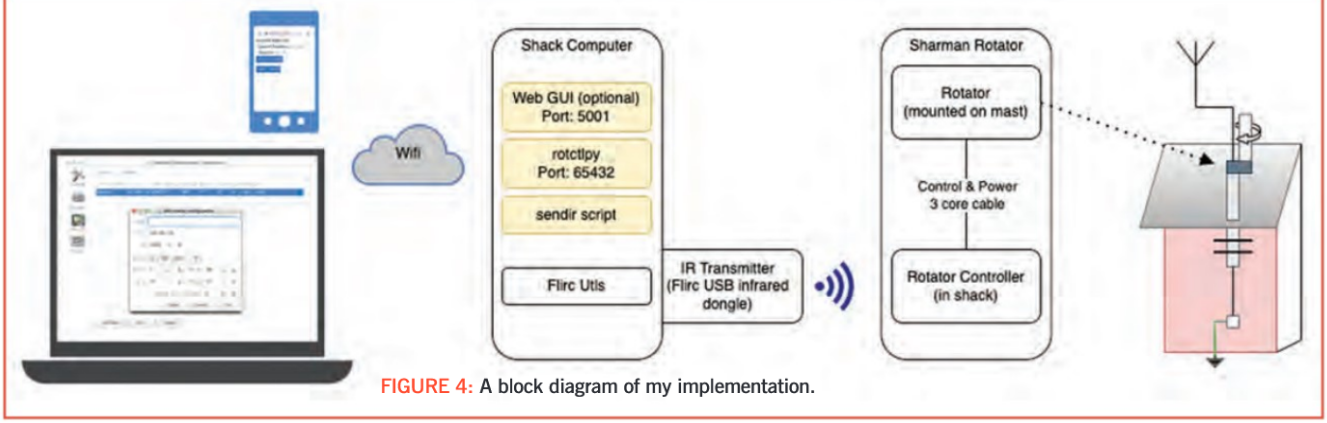


FIGURE 4: A block diagram of my implementation.

programmed, a single press of a button will rotate the antenna to the desired direction. The rotator is powered using a three-core cable connected to the controller; the cable is not supplied, but three-core mains cable may be used which is readily obtainable.

The instructions supplied with the rotator are useful, and there are a few YouTube videos that you can watch to give you an idea of how to set up the rotator and controller. There are, however, some necessary set-up tasks that must be undertaken for this implementation to work. After removing the controller from its packaging, it's best to program it and set it up before mounting the rotator on your mast. Follow the manual to see how to do this. Given that there are twelve programmable antenna directions, it makes sense to program the four cardinal directions along with two inter-cardinal directions between them. These are 0° (north), 30°, 60°, 90° (east), 120°, 150°, 180° (south), 210°, 240°, 270° (west), 300°, and 330°.

The rotator mount for the mast comes with a tongue that can slot into a groove cut into the mast, thus securing it and preventing undesired movement over time from wind and other factors. Having programmed the controller as suggested above, select 0° and mount the antenna to point towards the north.

At this stage, you should have a perfectly-good antenna rotator capable of rotating a small- to medium-sized antenna using the buttons on the main control unit or the infrared remote control.

Infrared transmitter

Computer control requires the computer to transmit the appropriate infrared codes, for which additional hardware is needed. There are several options available but I settled on the Flirc USB dongle (Figure 3) which is available for around £20 as an off-the-shelf solution. If you are feeling adventurous, it's entirely possible to substitute this dongle with an IR receiver and emitter diode, as well as the LIRC libraries, for a pound or so, and there are plenty of tutorials online showing you how to wire it all up.

```
+9090 -4401 +643 -453 +637 -456 +638 -456 +643 -452
+638 -456 +638 -456 +639 -456 +638 -458 +611 -1610
+639 -1588 +639 -1584 +642 -1585 +643 -1584 +638
-1585 +642 -1584 +639 -1588 +639 -456 +638 -1584 +669
-426 +612 -1615 +638 -456 +639 -456 +638 -1584 +643
-451 +643 -1584 +643 -451 +638 -1588 +639 -456 +648
-1578 +638 -1588 +638 -465 +630 -1584 +642
```

FIGURE 5: An example of a button sequence transmitted by the rotator remote control.

Flirc USB dongle

At first glance, the Flirc USB dongle was exactly what I wanted; however, I did struggle with both the software installation and the use. After installing the software, I needed to raise several support cases to get the implementation to work. This is all been sorted out now, so I hope that others will not have the same problems. Flirc USB software can be installed on Windows, macOS and

```
irtools sendir --raw="+9090 -4401 +643 -453 +637 -456 +638 -456 +643 -452 +638 -456
+638 -456 +639 -456 +638 -458 +611 -1610 +639 -1588 +639 -1584 +642 -1585 +643-1584
+638 -1585 +642 -1584 +639 -1588 +639 -456 +638 -1584 +669 -426 +612 -1615 +638 -456
+639 -456 +638 -1584 +643 -451 +643 -1584 +643 -451 +638 -1588 +639 -456 +648 -1578
+638 -1588 +638 -465 +630 -1584 +642" --repeat=1
```

FIGURE 6: Use of the flirc utils command to re-transmit the IR sequence.

```
Download Source code from https://github.com/stevendodd/rotctlpy/releases/latest
Extract zip file and open a command prompt in the new directory
pip install -r requirements.txt
python rotator.py
```

FIGURE 7: Installation commands.

```
python rotator.py --listen_port 5001 --host localhost --port 65432 -g
```

FIGURE 8: Invoking the optional web GUI.

Linux, and the first step is to install this software, along with the included 'flirc utils' package for command-line control.

If you are using the same hardware as I am, there is no need to undertake the capture and playback steps below, as the IR codes are already captured and implemented in my software; however, they are included here for completeness. The steps will obviously be different if you are using different hardware.

Once the Flirc USB software is up and running, you need to enable IR debug, point your rotator remote control at the Flirc dongle, and press a button on the remote control. You should be able to capture the output in the debug log; each button has a unique sequence, and you will need to do this for all 12 rotator memory positions. The output is in some proprietary timing format and will look a bit like that shown in **Figure 5**. Make sure to capture the full sequence of numbers, and label each set with the corresponding button.

Now that you have the raw data captured from the remote control, it is a simple task to re-transmit the signal using the flirc utils, using the command shown in **Figure 6**. This command should send the infrared signal and, if the rotator controller is in range, it should engage and rotate the antenna to the desired position.

FIGURE 9: The optional web GUI.

Rotator controller software 'rotctlpy'

The last piece of the puzzle is to create an implementation that can interface between the hardware and readily-available amateur-radio software. I decided to create a python3 implementation of HamLib rotctl, as most software that controls antenna rotators will support this interface.

The implementation is made up of three components indicated by the yellow boxes within the shack computer as shown in Figure 4. These are the rotctld-type server, a script to send IR commands, and an optional web GUI interface. The web GUI can be used from your mobile phone, or indeed from any other web browser. It was originally written by Mark Jessop, and I adapted it for my use case, and updated it for use with python3.

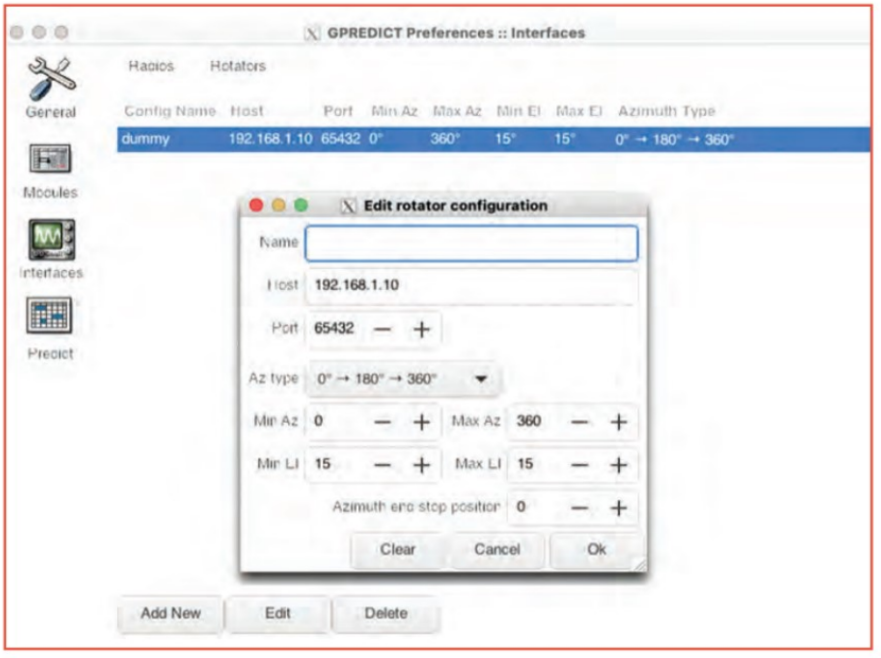


FIGURE 10: GPredict rotator configuration.

Once the server is running, it will accept rotctl commands on port 65432. If asked to rotate the antenna, it will invoke the script as a sub process to send the IR command. Please note that I have not tested it for use with multiple clients so, for example, if you are using GPredict and the supplied web GUI, you should only connect one client at a time to the server.

Installation

Having installed python3 on your system, the set of commands shown in **Figure 7** may be used to effect the following steps:

1. Download the rotctlpy source zip file, and the optional web-gui zip file, from https://github.com/stevendodd/rotctlpy/releases
2. Extract the release source zip file onto your file system.
3. Extract the optional web-gui zip file into the supplied, and unzipped, rotctld-web-gui folder made in step 2.

Running the implementation

In the top-level folder, there is a script called 'sendir.sh'. It takes a single command line option, namely the button you want to 'press' on the remote. You can invoke it as follows:

```
./sendir.sh A
```

This will transmit the sequence corresponding to button A on the remote control, using the Flirc command line 'utils' to trigger a rotation manually. If you are running on a UNIX-like operating system, this can be invoked via an SSH command remotely without having the rotctld-like server running. If you are using Windows, you can use the supplied batch file instead: 'sendir.bat'. If you use any other IR software, this script is the only place you will need to edit with your corresponding infrared commands.

The heart of the implementation is a rotctld python3 implementation. It can be started by simply running:

```
python3 rotator.py
```

Once invoked, the server will listen for a TCP/IP connection on port 65432, and will accept standard rotctl commands to operate, and report on, the rotator. Obviously, without an interface to the antenna controller, other than the ability to send an update-position IR signal, there is no way to guarantee that the antenna is indeed at a particular position. The

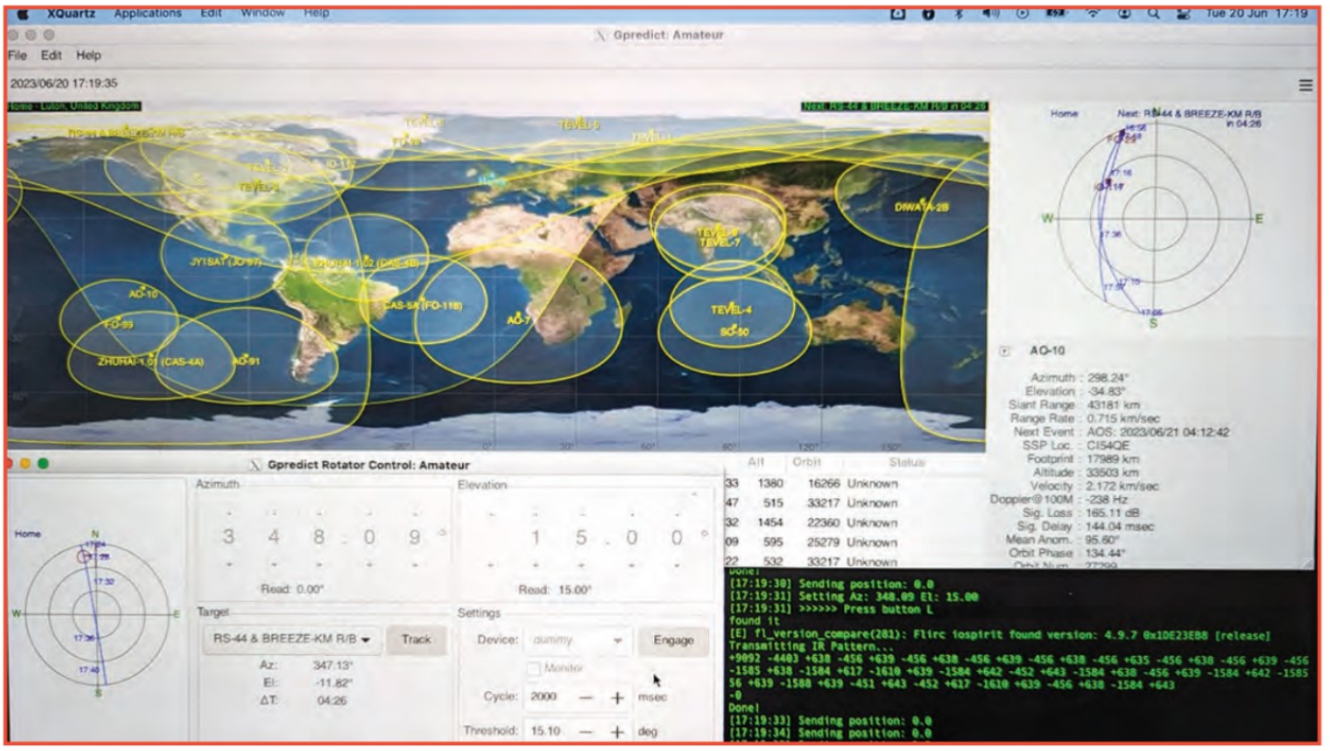


FIGURE 11: **Gpredict using rotctlpy to track satellites automatically.**

server will send an initialise command on start-up, and then attempt to maintain the antenna position in memory. If, for any reason, the position is incorrect, simply restart the server, which will in turn re-initialise the antenna.

Please note that this python script invokes the sendir script. If you are using alternative commands to transmit the IR signals simply update the relevant sendir script for your operating system.

Web GUI

I wanted a simple interface to the server using a browser and, rather than write my own, I included a previously-written rotctld web gui python implementation (see **Figure 9**). To start the GUI, simply move to the top-level directory, and invoke the command shown in **Figure 8**. The web GUI will then connect to the server, and you should be able to get access to it at the URL:

```
http://localhost:5001
```

I use my mobile phone to connect to my computer by substituting 'localhost' with the computer's IP address, which then allows me to operate the antenna via my Wi-Fi network from anywhere in the house. The GUI allows movement of the antenna incrementally 30° at a time through the full 360° rotation, or to reset to the initial position, and it reports the antenna position as registered in the server.

GPredict satellite tracking with rotator control

By now, you should have the ability to adjust the antenna manually by running the sendIR script locally or remotely, along with a web-based interface if you prefer. In addition, you should have a server running that can be connected to readily-available amateur-radio software, such as GPredict, and in this section I will show you how to connect GPredict to the server for automatic satellite tracking.

In the GPredict 'Preferences', you can set up an antenna rotator in the interfaces section (see **Figure 10**). Give your rotator a name which is used within the rest of the GPredict interface. In the host input box, add the IP address of the computer running the rotctlpy server. Select port 65432. The 'AZ type' should be set to 0° - 180° - 360°, with a minimum AZ of 0° and a maximum AZ of 360°. Set the minimum and maximum EL to 15; this is not relevant as the elevation is fixed, but my script by default sends back 'in EL of 15°'. Save your rotator, and exit the preferences.

To track a satellite, open the rotator-control dialogue box within Gpredict, select a target satellite, and press the 'track' button. In the settings section of the rotator-control dialogue box, select the name of the rotator you set up on the previous step. Choose and a threshold of 15.1°, and a cycle of 2000ms; this will update the antenna position every two seconds. The threshold indicates when GPredict needs to send an updated target position to the rotctl server. Since our pre-programmed settings are 30° apart, 15° is the halfway point which will trigger movement to the next predefined direction.

Once these settings have been applied, press the 'engaged' button to connect to the rotctlpy server; you should see the antenna position plotted on the satellite polar view, along with the reported assumed position, whenever GPredict asks the antenna to rotate. See **Figure 11**.

Improvements

Future enhancements could include adding a solar-powered compass to the rotator mast, adding the ability to obtain true directional readings. The Raspberry pi zero 2 W, with the 'Sense Hat' that was for developed for use on the international space station (https://www.raspberrypi.com/products/sense-hat), would be a good choice. I have also integrated the solution into 'Home Assistant' with a clickable graphical compass as a replacement for the web GUI.

This implementation is not a commercially-polished application, and requires a little bit of tinkering to get working. However, it works a treat and will provide you with low-cost integrated computer antenna control. Have fun!

Feature

The M17 project

FIGURE 1: OpenHT, an open-source SDR-based handheld transceiver.

FIGURE 2: The Module 17 modem.

This is an open-source digital voice standard which promotes innovation, and which should attract young 'hardware hackers' to amateur radio.

Introduction

The freedom to design, build, and experiment has always been an important part of amateur radio. As our hobby migrates to digital voice technology, that freedom may be limited by proprietary technology used to code and decode the signals. What experimenters need is an open-source digital standard, ie a standard that anyone can use without having to buy a licence or pay royalties, and that anyone can extend or modify provided that they share their code with the amateur-radio community.

Digital voice (DV) in the UK's VHF and UHF bands is currently served by no fewer than seven different standards: System Fusion from Yaesu (YSF), digital mobile radio (DMR) from the European Telecommunications Standards Institute (ETSI), D-STAR from Icom, next-generation digital narrowband (NXDN) from Icom and Kenwood, project 25 (APCO P25) developed by public safety professionals in North America, terrestrial trunked radio (TETRA) developed by public safety and two-way radio industry experts together with ETSI, and M17 developed by a community of open-source developers and radio enthusiasts. Four of these standards use the proprietary advanced multi-band excitation (AMBE+2) codec (a codec is a device that converts analogue voice signals to digital signals for transmission, and digital signals to analogue voice signals on reception), designed and licensed by Digital Voice Systems Inc (DVSI). D-STAR and TETRA use older codecs: D-STAR uses AMBE and TETRA uses algebraic code-excited linear prediction (ACELP) that, while beyond its 20-year patent protection period, still contains proprietary elements. M17 uses the completely-free and open-source codec 2, developed specifically for amateur radio. A comparison between the various standards is shown in **Table 1**.

It's important to keep all roads to innovation open. Amateur radio operators have made significant contributions to the technology over the years. For instance, Louis Varney, G5RV invented his popular multiband antenna, and Edwin H. Armstrong, W2MXN invented FM radio.

Amateur radio enthusiasts, who hope to make contributions to digital voice technology, may encounter obstacles. When equipment manufacturers decided to offer digital voice, they had two options. They could develop codecs for amateur radio from scratch, not a trivial task, or they could license codecs already developed for commercial and public-safety markets. For reasons that made good business sense at the time, they decided to license existing technology. However, this created a couple of problems for home-brewers and experimenters. It meant that anyone making their own interoperable device would have to purchase a codec from a licensed manufacturer. That may not sound like a big imposition, but it limits the home-brewer's design choices. And anyone wanting to customise a commercially-manufactured radio might be prevented from doing so if the codec is implemented in firmware. The manufacturer might not care if it's just one radio, but what if that person decides to share their code by publishing it online?

There is another argument for developing codecs specifically for amateur radio. Although the licensed codecs are the result of years of development and experience in commercial and public-safety markets, they aren't optimised for amateur radio applications, which are exceptionally diverse and continue to evolve. Free and open-source codecs would give the amateur radio community greater control over its future.

M17 is a reasonable solution, both in terms of ensuring experimenters are free to experiment, and providing a way forward. The M17 specification, source code, hardware designs, and algorithms are all open. The software is licensed under the GNU general public licence version 2.0, and the hardware under the Tucson amateur packet radio (TAPR) open hardware licence. There is also a management system to facilitate sharing of source-code additions and changes.

There is one more problem that M17 might be able solve. Because there are several different standards, radios can only talk to repeaters and other radios supporting the same standard. The market for VHF/UHF amateur radios is, therefore, quite fragmented. An open-source standard could serve as a vendor-neutral bridge between the standards, by enabling dual-mode mobile radios, handheld transceivers, hotspots, and repeaters (M17-YSF, M17-DMR, etc). This would promote greater interoperability, and would facilitate the long-term development of a universal digital voice standard for amateur radio, all while continuing to protect the manufacturers' investments in current standards.

Enter the M17 project

The M17 project was founded by Polish radio amateur Wojciech Kaczmarski, SP5WWP in 2019. The name 'M17' comes from the street address of a radio club in Warsaw (Mokotowska 17). After obtaining his licence in 2016, Kaczmarski began experimenting with Codec 2, the open-source codec developed by David Rowe, VK5DGR for FreeDV, some software for using digital voice in the HF bands. Kaczmarski wanted to create an alternative to the popular DMR standard, because the DMR specification is difficult to understand, and its proprietary codec inhibits 'hacking', a term many people use for building and modifying hardware. The effort attracted support, the project grew, and in 2021 Kaczmarski received the ARRL Technical Innovation Award for developing the M17 protocol.

The goal of the M17 project is to ensure that the amateur radio community has the freedom to build, learn, and innovate by creating and evolving a digital voice/data protocol specifically for amateur radio. M17 supporters believe that it will lead to more interesting products and that, in turn, will make amateur radio more attractive to young people.

Enthusiasm for designing, tinkering, and inventing is very much alive in the amateur-radio community. Recent examples include: multi-mode digital voice modem for hotspots and repeaters (MMDVM), high-performance software-defined radio (HPSDR), experimental firmware for select radios (OpenRTX), and multi-platform software for digital modes like PSK31 (Fldigi) and FT8 (WSJT-X). The M17 Project is driving the development of OpenHT, an open-source SDR-based handheld transceiver that operates in the 440MHz and 2.4GHz bands, and that supports a variety of modulation schemes (including AM, FM, SSB, 4FSK, and SSTV). OpenHT will enable amateur radio enthusiasts to build and customise small handheld transceivers (see **Figure 1**).

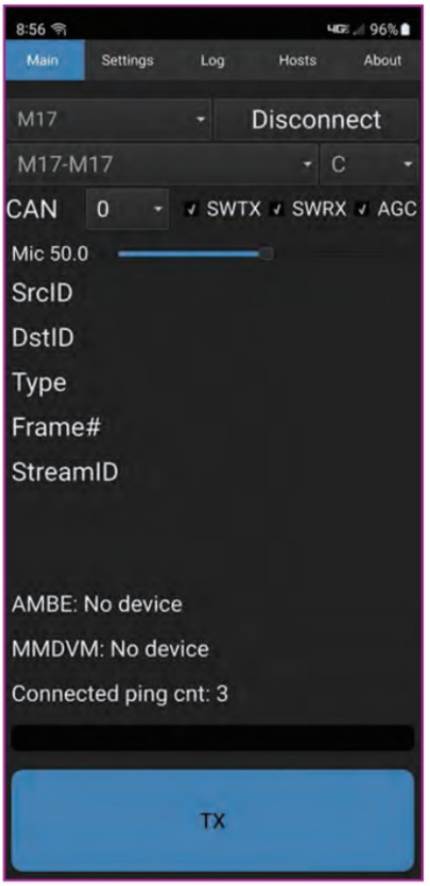


FIGURE 3: **Droidstar digital voice client software.**

Though proprietary technology has its place (for example the Apple computer), independent developers and experimenters tend to prefer open systems. Open systems drew many people to TCP/IP and Linux. TCP/IP triumphed because it was just good enough, easy to implement, and not proprietary. Linux provided an alternative to operating systems controlled by large corporations such as Microsoft and Apple.

M17 is already attracting young people; most M17 hackers are in their 20s or 30s. The M17 specification is relatively easy to understand, and can even be used as a digital communication teaching tool. M17 has also attracted financial grants: Amateur Radio Digital Communications (ARDC), a private foundation supporting amateur radio technology development, awarded the M17 Project $250,000 in 2021 and an additional $228,900 in 2022.

The M17 specification

The M17 specification can be viewed and downloaded at **[1]**. The specification defines three network layers: the physical layer, the data-link layer, and the application layer. The M17 protocol is versatile, supporting voice, point-to-point data, and telemetry. The M17 specification is also extensible so new capabilities may be added over time.

The physical layer describes the air interface: how ones and zeroes are represented over the air using 4-level frequency-shift keying (4FSK), the number of bits per symbol (2), and the symbol rate (4,800 symbols per second times 2 bits per symbol = 9,600 bits per second). M17 uses channels of width 9kHz with 12.5kHz spacing. Each transmission includes a preamble (to prepare the receiver), a synchronisation burst (to align the clock), the data payload, and an end-of-transmission marker.

The data-link layer describes how the data is organised into the four types of frames (link setup, bit error rate test, stream mode, and packet mode), and the error correction method. The link setup frames include the source and destination addresses. The bit error rate test frames are used to test hardware, and ensure interoperability between disparate devices. The stream frames are used to send continuous data such as voice. The packet frames are used to send data in packets of up to 823 bytes each. The throughput ranges from 3kbps to 4.7kbps, depending upon the packet size.

The application layer describes how various tasks are accomplished. For instance, it defines parameters for audio streaming using Codec 2, and the protocol (such as automatic packet reporting system (APRS), AX.25, etc) when transmitting packet data. Codec 2 offers good audio quality at low bit rates (1,600 or 3,200 bps). The application layer also defines parameters specific to user devices (clients), repeaters, and gateways (such as internet reflectors). M17 is designed to support IP networking for repeater linking and communicating via internet reflectors. M17 also supports strong encryption for use where legal (eg in Poland).

M17 development projects

There are currently several efforts underway to develop M17-capable hardware, as well as to adapt M17 to existing hardware. Repositories of software code for many of the following projects can be found at Github (see **[2]**). M17 community participants can also be found chatting on Discord **[3]**.

Module 17 **[4]** is a modem that sits between the microphone and radios capable of running 9600bps, such as the Kenwood V71 and D710, and the Motorola CDM series. Module 17's baseband output feeds into the radio's packet data input. See **Figure 2**.

The popular Digirig computer interface **[5]** can now be used to run M17 from a PC, using software and a soundcard, with radios such as the

Ira Brodsky, KC9TC
ibrodsky64@gmail.com

Yaesu FTM-200, FTM-300, FTM-400, and FTM-500. The Digirig cable is connected to the packet-data input and packet-data output pins.

Mobilinkd's TNC4 **[6]** is a hardware-based solution for adding M17 to a radio. It is an open-source TNC platform, and is also available in a bread-board configuration (NucleoTNC). The TNC4 and NucleoTNC provide alternatives to the PC-plus-soundcard approach.

The OpenRTX project **[7]** develops free and open-source firmware for digital amateur radio. The goal is to enable amateur radio operators and experimenters to customise and add features (such as voice prompts, different languages, and M17 support) to existing radios. Note that this project is being pursued independently of the manufacturers; changing the radio's firmware is carried out at the owner's risk, although users are encouraged to keep a backup copy of the factory firmware, which in most cases is easy to reinstall. OpenRTX could enable upgrading a DMR radio to a dual-mode DMR/M17 radio (assuming that there are no legal obstacles). Radios used in the OpenRTX project currently include Tytera MD-380/390, MD-UV380/390, and TYT MD-9600, Radioddity GD-77, and Baofeng DM-1801.

Multi-mode digital voice modem (MMDVM) **[8]** is an open-source hardware/software modem, developed by Jonathan Naylor, G4KLX that supports all amateur-radio digital voice standards, including M17. It is widely used in hotspots and repeaters. It runs on platforms that use ARM Cortex-M3, M4, or M7 processors, such as the Arduino Due. MMDVM_HS is José Andrés Uribe's (Andy, CA6JAU) hotspot version. MMDVM_CM is Doug McLain's, AD8DP solution for adding cross-mode capability to internet reflectors (eg M17 to YSF). M17Client, also by Jonathan Naylor, adds M17 to MMDVM modems and hotspots, and includes features such as voice, simultaneous text messaging, and GPS locating data.

WPSD **[9]** is a digital voice management suite of software for hotspots and repeaters that supports M17, DMR, D-Star, YSF, P25, and NXDN, and runs on the Raspberry Pi. The name is an acronym for 'W0CHP-PiStar-Dash', and is derived from the author's popular Pi-Star software. WPSD works with MMDVM, but doesn't do any encoding or decoding; it just takes data from the radio-modem combination and sends them over the network, and takes data from the network and sends them back to the radio via the modem.

Mvoice software **[10]** runs on Linux and can be used to transmit and receive M17 over the internet. A user with just a headset and a Raspberry Pi (or other Linux device) can use it to connect to an M17 reflector, and chat with other M17 users.

Mrefd **[11]** is an M17 reflector program that supports up to 26 channels. M17 clients (such as devices running Mvoice), M17 hotspots, M17 repeaters, and other M17 reflectors can be linked to each channel. The voice stream from each client is heard by all other clients on that channel. Mrefd runs on Linux (Debian or Ubuntu distribution recommended).

DroidStar **[12]** is digital voice client software that runs on multiple platforms (including Android, iOS, Linux, Windows, and MacOS) and supports multiple digital voice protocols (see **Figure 3**). It comes with M17 built-in; additional protocols require a plug-in voice encoder (vocoder). Droidstar can be used to talk directly over the internet or a radio and a modem can be added to turn a mobile phone into a hand-held transceiver.

M17 is also being integrated with software defined radio platforms. SDR++ software for Windows, MacOS, and Linux **[13]** includes a built-in M17 decoder. OpenWebRx, a web-based multi-user receiver **[14]** that can be operated from any web browser, will also support M17.

M17 and the future of amateur radio

In an increasingly digital and internet-connected world, it's essential that radio amateurs continue to develop their digital voice/data capabilities. An open-source standard is needed so that the entire amateur radio community, licensed operators as well as equipment manufacturers, can contribute to the process. M17 appears to be well-positioned to help make that happen.

References

[1] https://spec.m17project.org
[2] https://github.com/M17-Project
[3] https://discord.gg/G8zGphypf6
[4] https://github.com/M17-Project/Module_17
[5] https://digirig.net/
[6] https://store.mobilinkd.com/products/mobilinkd-tnc4
[7] https://openrtx.org/#/
[8] https://github.com/g4klx/MMDVM
[9] https://w0chp.radio/wpsd/
[10] https://github.com/n7tae/mvoice
[11] https://github.com/n7tae/mrefd
[12] https://github.com/nostar/DroidStar
[13] https://github.com/AlexandreRouma/SDRPlusPlus
[14] https://www.openwebrx.de/

Table 1: M17 compared with the other VHF/UHF digital voice standards used by radio amateurs in the UK.

	M17	System Fusion	DMR	D-STAR	NXDN	APCO P25	TETRA
Approx. number of Repeaters, UK	9	373	318	209	57	53	7
Developer/Vendors	Amateur radio community/ multivendor	Yaesu/Yaesu	European Telecomms Standards Institute/ multivendor	Japan Amateur Radio League/ Icom, Kenwood, and Flex	Icom and Kenwood/ multivendor	Association of Public Safety Comms Officials/ multivendor	European Telecomms Standards Institute/ multivendor
Type of standard	Free and open source	Proprietary	Partially open	Partially open	Partially open	Partially open	Partially open
Codec	Codec 2 (open source)	AMBE+2 (patent-protected)	AMBE+2 (patent-protected)	AMBE (copyright-protected)	AMBE+2 (patent-protected)	AMBE+2 (patent-protected)	ACELP (codebook-protected)
Modulation	4FSK	4FSK	4FSK	GMSK	4FSK	4FSK	DQPSK
Data features	Callsign, text messages, location, images	Callsign, text messages, location, images	Callsign, text messages, location	Callsign, text messages, location, images	Callsign, text messages, location	Callsign, text messages, location	Callsign, text messages, location
Advantages	Fully open source, created for amateur radio	Largest installed base of repeaters, WIRES-X	Well supported, talkgroups	Created for amateur radio, up to 128 kbps data	Can use 12.5 or 6.25 kHz channels	Designed for emergency communication	Supports four simultaneous conversations in 25 kHz
Disadvantages	Limited installed base (new entrant)	Single vendor	Designed for commercial use, difficult to set codeplug	Inferior codec, expensive	Designed for commercial & public safety, limited installed base	Mainly used gear for amateur use, not all models suitable for amateur radio	Equipment for amateur use limited to 430 MHz band

HF absorption in winter

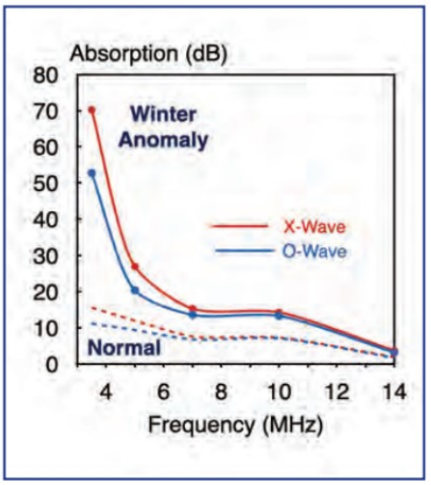


FIGURE 1: Computer simulations of 1000km transmissions at midday from London in January 2024. Absorption loss (dB) is shown *vs* frequency (MHz) during peaks in a strong winter absorption event and during normal winter conditions.

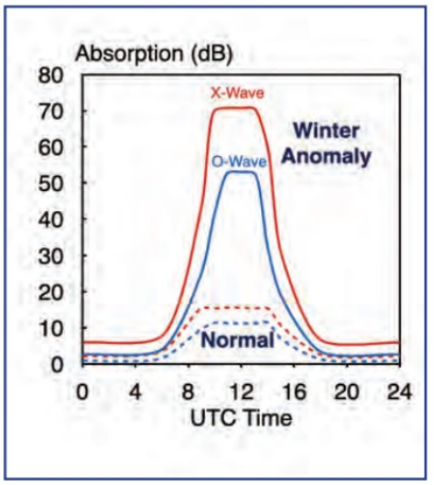


FIGURE 2: Simulated absorption loss (dB) of 3.5MHz signals *vs* UTC time during a strong winter absorption event compared to normal winter conditions.

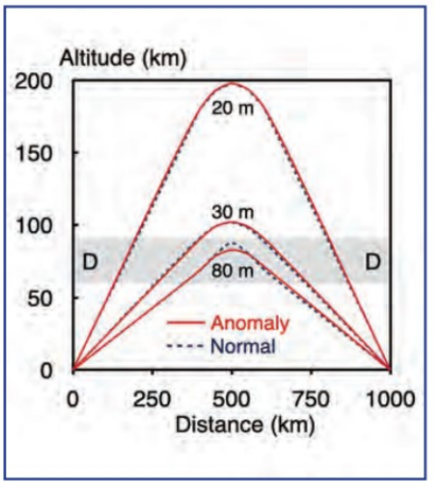


FIGURE 3: Altitude (km) *vs* distance (km) for O-waves during 1000km transmissions on the 80, 30 and 20m bands. The grey-shaded area represents the D-region.

On some winter days at mid-latitudes the absorption loss from HF signals is unexpectedly high.

Received signal strength can vary erratically for more than a week during one of these events. These variations are called winter anomalies. Measurements using rockets and high-altitude balloons in the 1975-6 Western Europe Winter Anomaly Campaign showed that these anomalies are from enhancements of the D-region (altitude 60-90km). They are caused by seasonal variations in the neutral atmosphere **[1]**. The horizontal extent of an affected patch in the ionosphere can be anywhere from hundreds to thousands of kilometres.

Figure 1 shows computer simulations of 1000km transmissions from London in January 2024, during peaks in a strong winter anomaly **[2]**. Absorption (dB) *vs* frequency (MHz) is shown for midday transmissions. Each data point represents a band from 20m to 80m. In the ionosphere, linearly-polarised signals split in two different waves with polarisations that rotate in opposite directions – ordinary (O) waves and extraordinary (X) waves. X-waves are more strongly absorbed. Solid lines in Figure 1 show absorption during a strong anomaly, and dashed lines show normal propagation. Absorption caused by the anomaly is very high at lower frequencies.

HF absorption in the D-region is greatly reduced at night. **Figure 2** shows how simulated absorption loss for 3.5MHz signals (1000km range) varies with UTC time during a strong anomaly (solid lines) and during normal conditions (dashed lines). At midday, when the Sun is at the highest point in the sky, absorption during the anomaly is at the levels shown in Figure 1. When the Sun is lower in the sky, absorption is closer to normal levels.

In these examples, the winter anomaly does not affect the range of transmissions. **Figure 3** shows altitude (km) *vs* distance (km) for O-waves during 1000km transmissions on the 80, 30, and 20m bands. The grey-shaded area represents the D-region. Solid lines show paths during peaks in a strong winter anomaly event, and dashed lines show normal propagation. Transmission elevation angles for a 1000km path are the same during the anomaly as during normal propagation. This is true even for the 80m band, where the anomaly changes the signal path in the D-region. Graphs of altitude *vs* distance for X-waves show similar results.

The term 'winter anomaly' is also used to describe seasonal increases in maximum usable frequencies (MUF) compared to summer **[3]**. The MUF anomaly increases your opportunities to lower HF attenuation during winter absorption events by moving to a higher frequency. For example, the *actual* midday MUF (1000km) for the conditions in Figure 1 was 17MHz. This is based on a measurement of the critical frequency foF2 at the Dourbes, Belgium, ionosonde on the date used in the simulations (15 January 2024) **[4]**. Simulations with a simplified ionosphere model in Proplab provide useful insights about propagation, but they are not intended for accurate propagation predictions.

References

[1] D. Offermann, A winter anomaly campaign in Western Europe," Phil. Trans. R. Soc. Lond., A 296, 261-268, 1980: https://www.jstor.org/stable/36447

[2] Proplab-Pro V3.2, Solar Terrestrial Dispatch:. http://spacew.com/index.php/software

[3] Peter DeNeef, AE7PD, *RadCom*, April 2024, p. 66.

[4] Jim Bacon, G3YLA, PROPquest: https://www.propquest.co.uk/graphs/php

Peter DeNeef, AE7PD
HamRadioAndVision@gmail.com