RFZero based Mixed Mode 2m beacon project

Note: this presentation documents a work in progress, not a completed project….

- Why beacons: propagation indication, receiver verification
- How should a beacon be operated (power out, mode, 24/7, frequency stable)
- **Power**: limited to 100 watts output by FCC rule (**§97.203 Beacon station, https://www.arrl.org/part-97-text)**
- **Mode**: traditional carrier with CW or AFSK I.D., until the advent of digital modes, now we have mixed mode or Machine Generated Mode (traditional analog + digital mode) beacons becoming normal
- Beacons should be on the air 24/7; otherwise, how useful is the beacon?
- Beacons should be frequency stable so that they are predictable and can be used for a frequency reference on the band they occupy

How I became involved in this project:

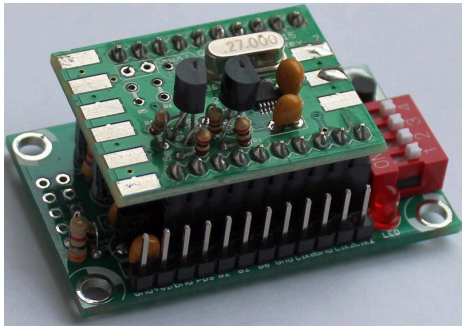
In the early to mid 2000s, I was given a set of four VHF+ (6m - 70cm) beacons to use by Bert Rollen, K4AR. Bert had the beacons on a mountain top in eastern Tennessee until he lost access to the site, Additionally, he grew weary of maintaining the beacons due to equipment failures, usually the power supply. At the time, I had access to an old AT&T long lines site near Crossville, TN, so some friends and I installed the beacons on that site. All was well until the site was sold and I was told to move out in 2009. Several years later, another friend asked about the feasibility of putting a 6m beacon on the air. I had some old G.E. MASTRII low band mobile transceivers and I started experimenting with using one as a beacon. That didn't work out well at all, but along the way I learned about the Si5351A synthesizer as used in the QRP Labs ProgRock:

What is the Si5351A frequency synthesis chip
https://cdn-shop.adafruit.com/datasheets/Si5351.pdf

In short, it is a I^2C programmable (almost) any frequency CMOS clock generator + VCXO

Original ProgRock - triple GPS-disciplined programmable crystal from QRPLabs.com

This kit is a simple minimalist controller for the Si5351A Synth kit (included with the ProgRock kit). It is intended as a programmable crystal replacement. It has three independent outputs with frequency range 3.5kHz to approx 300MHz, and can be optionally GPS disciplined



The features of this kit are as follows:

- Includes Si5351A Synth kit AND "ProgRock" PCB kit
- **Easy construction, no surface-mount components to solder (Si5351A already pre-soldered)**
- 3 independent 3.3V p-p squarewave outputs (2 if you use GPS discipline)
- **You can feed the outputs through LPF kits to get sinewave outputs**
- 8 selectable “banks” of frequencies, chosen by 3 input control signals
- **Frequency range 3.5kHz to 200MHz**
- Extended frequency range up to approx. 300MHz if you don’t mind violating the Si5351A datasheet specifications
- Quadrature output mode (Clk0 and Clk1 on same frequency but configurable 0, 90, 180 or 270-degree phase offset)
- GPS frequency discipline using 1pps from a GPS receiver
- Unique power supply noise filter circuit designed by Alan Gray G8LCO
- Power supply voltage 5V, or using LM317LZ regulator (supplied), range is 5-12V DC
- Minimalist configuration user interface using 4-way DIP switch, push button, and LED
- Optional serial port configuration (requires firmware pr1.01 and above)
- Frequencies and configuration stored in non-volatilte memory (EEPROM) for next power-up

Some (several) years later, the Arduino microcontroller became very popular among builders of many things electronic, programmable & controllable

Arduino microcontroller overview https://www.arduino.cc/en/Guide/Introduction
Arduino is an open-source electronics platform based on easy-to-use hardware and software. Arduino boards are able to read inputs - light on a sensor, a finger on a button, or a Twitter message - and turn it into an output - activating a motor, turning on an LED, publishing something online. You can tell your board what to do by sending a set of instructions to the microcontroller on the board. To do so you use the Arduino programming language (based on Wiring), and the Arduino Software (IDE), based on Processing.

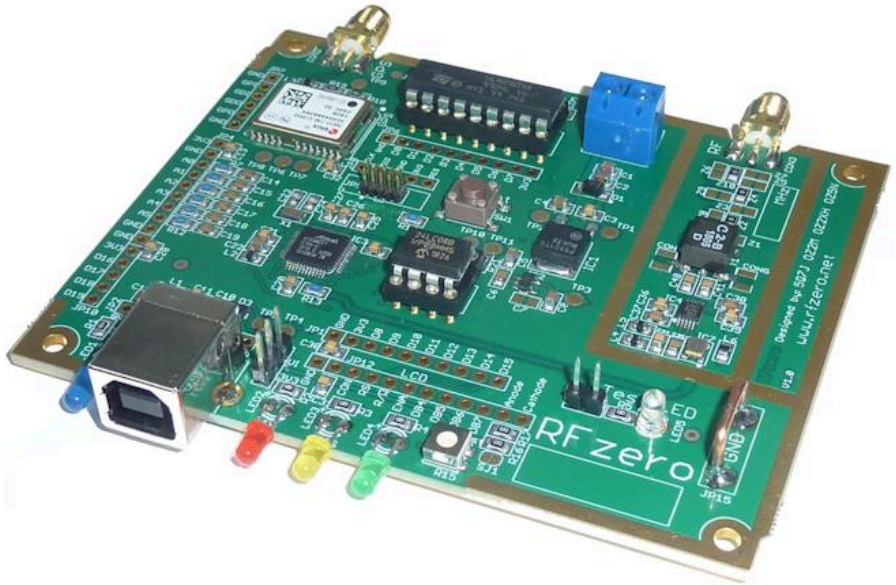
What is Arduino Zero:
The Zero is a simple 32-bit extension of the microcontroller platform established by the UNO. The Zero board expands the family by providing increased performance, enabling a variety of project opportunities for devices, and acts as a great educational tool for learning about 32-bit application development. The Zero applications span from smart IoT devices, wearable technology, high-tech automation, to crazy robotics. The board is powered by Atmel’s SAMD21 MCU, which features a 32-bit ARM® Cortex® M0+ core. One of its most important features is Atmel’s Embedded Debugger (EDBG), which provides a full debug interface without the need for additional hardware, significantly increasing the ease-of-use for software debugging. EDBG also supports a virtual COM port that can be used for device and bootloader programming.

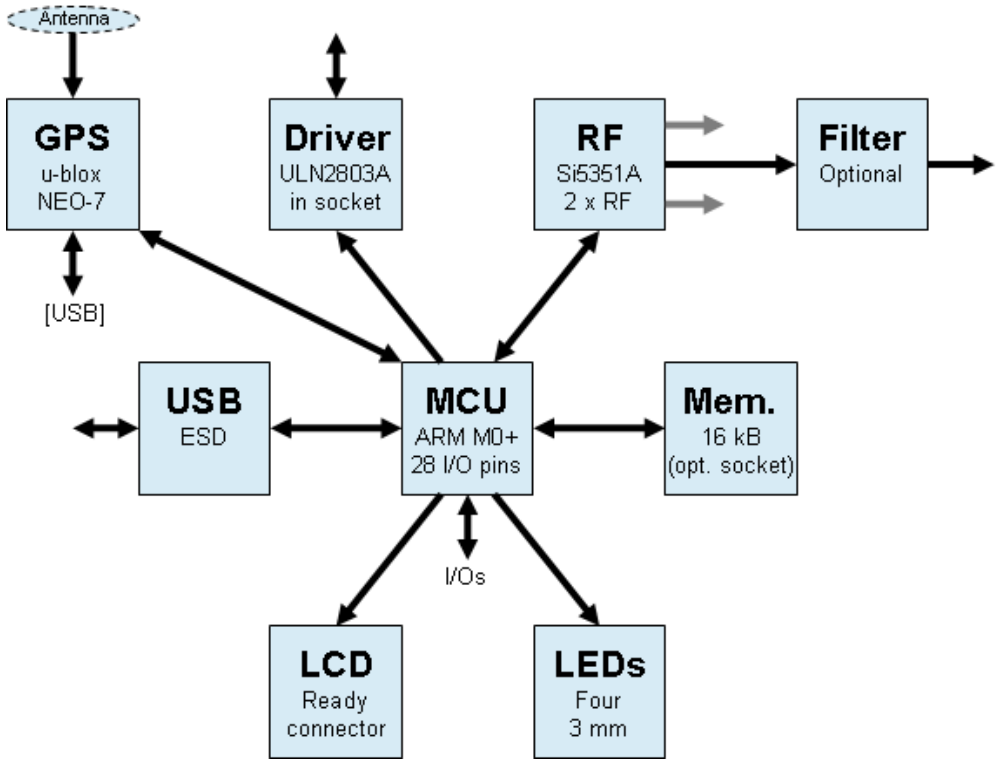
Warning: Unlike most Arduino boards, the Zero runs at 3.3V. The maximum voltage that the I/O pins can tolerate is 3.3V. Applying voltages higher than 3.3V to any I/O pin could damage the board.

What is RFzero:
Putting the three together, the RFzero team of **Johan, 5Q7J, Bo, OZ2M, Hans, OZ2XH & Steen, OZ5N** have created the RFzero multi-function oscillator, giving experimenters a powerful platform for building and experimenting with RF circuits and programming within the Arduino realm.
The RFzero™ can generate frequencies from 2289 Hz and close to 300 MHz. At the same time 28 I/O pins are available and eight of those are via an ULN2803A power driver. The programming and configuration are done via the USB port, thus an external hardware programmer isn't needed.

The RFzero™ has been developed for radio amateurs, RF enthusiasts and everyone else who wants to extend their Arduino skills in combination with RF. The RFzero™ board is largely compatible with the Arduino Zero and Arduino M0+ boards. However, it has been carefully designed for flexible use in RF applications and with attention to the frequency spectrum and stability performance. Many users say that the RFzero™ is the best Si5351A RF solution available.

The RFzero & functional block schematic:





Using the RFzero you can learn how to program an RF device and other electronic circuits or you can learn about RF circuits if you are already skilled in software development. The setup is very flexible, you can roll your own programs from scratch or you can use the RFzero library and programs together with the Arduino Integrated Development Environment (IDE).

The RFzero key features are

- GPS
    - u-blox NEO-7 GPS receiver with world leading documentation
    - On-board pre-amplifier supply via coax
    - SMA female input
    - On-board header for GPS data access from other units
    - Optional back-up from battery or super cap.
    - Optional USB port directly to the GPS receiver e.g. for computer time keeping
- RF
    - Si5351A clocked at 27 MHz and measured continuously for maximum accuracy
    - Fundamental frequencies from 2289 Hz and close to 300 MHz
    - Harmonics can also be used with reduced spectrum performance, e.g. WSPR on 1,3 GHz is possible and for PI4 on 10 GHz via x96 multiplier
    - 1 mHz frequency resolution
    - Push-pull coupled outputs for more output and lower even harmonics, or
    - Combined outputs for two-tone signals, or
    - Quadrature I/Q outputs via optional U.FL sockets
    - Lowest phase noise/time jitter
    - On-board pads for custom filter
    - On-board RF decoupling
    - RF section can be further shielded
    - Separate voltage regulator for the RF section
    - Four layer PCB for optimum RF performance
    - SMA female output
    - 13 dBm output 400 kHz to 200 MHz
- Micro controller
    - Arduino compatible (Arduino Zero and Arduino M0+)
    - 32 bits ARM M0+ processor running at 48 MHz and with 32 kB SRAM

    - 16 kB EEPROM
    - Optional EEPROM in tulip socket
    - USB port, with ESD protection, for programming and controlling purposes
    - ARM Cortex connector, Four SPI ports, Four I2C/Wire ports, Five serial ports
    - One Real Time Clock
    - Four status LEDs (on, TX, GPS PPS and GPS valid)
    - On-board LCD connector for either 3,3 V or 5 V LCD with contrast and backlight control
    - 28 I/O pins
    - Up to eight 12 bits analog inputs
    - One 10 bits analog (DAC) output
- Separate voltage regulator for the digital circuitry
- Darlington power driver IC ULN2803A in tulip socket e.g. for controlling relays
- Possibility to add a shield for additional functionality or expansion
- On-board test LED
- On-board ground loop for alligator clip
- 11 dedicated test points for easy troubleshooting
- Half Eurocard PCB size that fits into a variety of boxes including a standard metal sheet box for best RF shielding
- Power supply via USB or 5 V to 9 V to a terminal block
- Gold immersion board for best assembly quality and life time performance

What can the RFZero be used for?

What's this mixed mode beacon business about? (https://www.oz7igy.dk/pi4/)

PI4: The name Pharus Ignis 4 comes from the ancient words for beacon, lighthouse and fire - Pharos (from Greek to Latin pharus and coming from the Lighthouse of Alexandria), Ignis (Latin: fire) and 4 for the four FSK tones.

Why PI4

The reason for using PI4 + CW + carrier for the OZ7IGY beacons is:

- The analog part of the identification must be frequent both to identify the beacon but also to "handle" QSB
- The beacon must be readable even when the path is distorted i.e. by rain scatter, aurora etc. Who says we have finished detecting new propagation techniques? So the beacon must also be "forward compatible"
- It must be possible to decode the MGM part even if the receiver is not frequency locked
- Today most beacons identify themselves every 30-45 seconds. But waiting for the identification "always" seems too long; like waiting in a telephone queue always seems too long, the perceived time when waiting always seems to be to long
- A 2 min sequence is to long; the shorter, the better
- Identical sequence every minute for reliability/predictabilty
- It should be possible to calibrate to the beacon. Thus a carrier is needed to zero-beat on. Today this is specified to be about 20-30 seconds
- The analog identification should be readable by "all of us" not just the very high speed CW operators. Thus 12 WPM/60 LPM as already specified seems to make sense
- Must fit into the current 1 kHz beacon to beacon spacing structure
- The **MGM** must be transmittable via a class C amplifier to save power

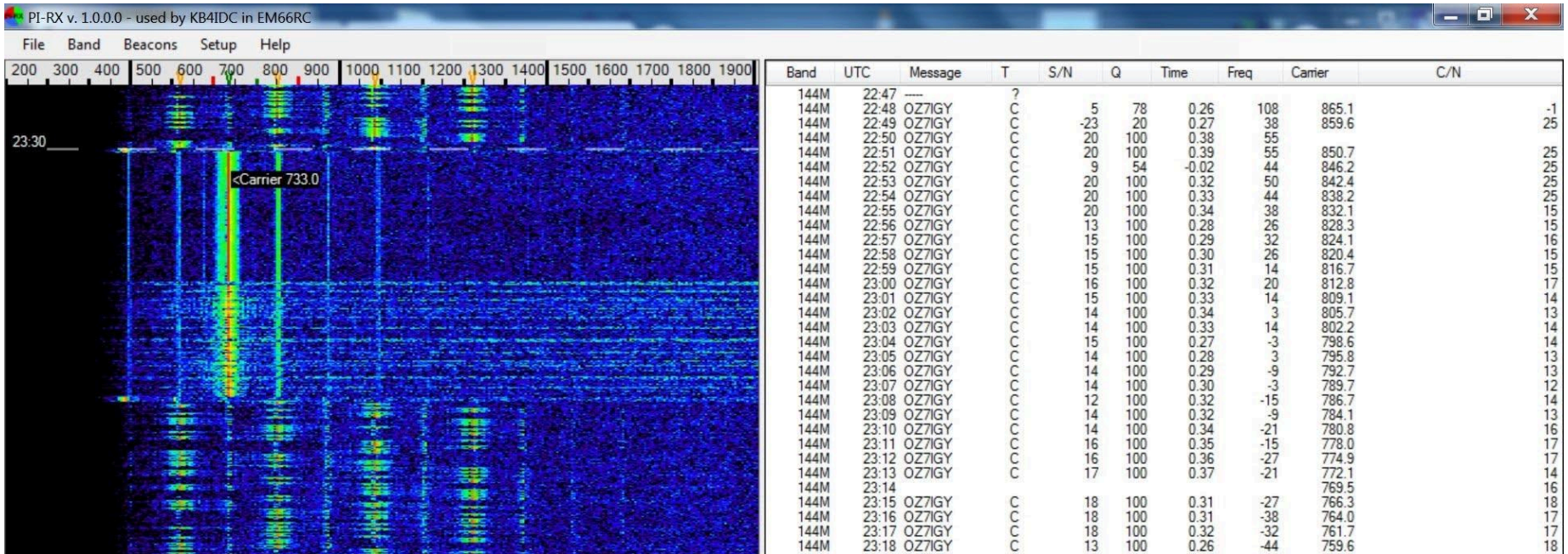
PI4 is specifically designed with beacons and V/U/SHF propagations in mind. It is far more robust to path irregularities and equipment inaccuracies/an e.g. WSPR and JT9 that both also would require a long sequence. JT65 modes are in-between WSPR/JT9 and PI4, and JT4, when it comes to robustness but cannot fulfil an identical sequence every minute without other sacrifices unless using the JT65B2/C2 submodes. The JT4 submode JT4G

has better path robustness than PI4 but takes about twice as long to transmit thus it will not fulfil the sequence requirements. PSK2k, FSK441, JT6M, JT9-fast, MSK144 and ISCAT are all “fast modes” but are not as sensitive as the other modes. Because of this is PI4 the MGM used by the OZ7IGY Next Generation Beacons.

Decoding PI4

The OZ7IGY PI4 beacons transmit in a 1 minute cycle starting at the full minute. The colored line below illustrates the mixed mode sequence where P indicate pauses. The CW is sent at 60 LPM/12 WPM.

First it sends PI4 (call sign, **M**achine **G**enerated **M**ode) over a 24.33s period, a subsequent pause of .63s, then 25s of CW identification (call sign and locator), followed by a 500ms pause, then 10s of carrier until 59.5s followed by a 500 ms pause until the end of the cycle. The resulting waterfall and decoding in PI-RX will look like the screen dump below. By registering the beacon in the database



(https://www.iaru-r1.org/about-us/committees-and-working-groups/vhf-uhf-shf-committee-c5/vhf-beacon-coordination/), PI-Rx can display the location of the beacon from the decoded MGM callsign.

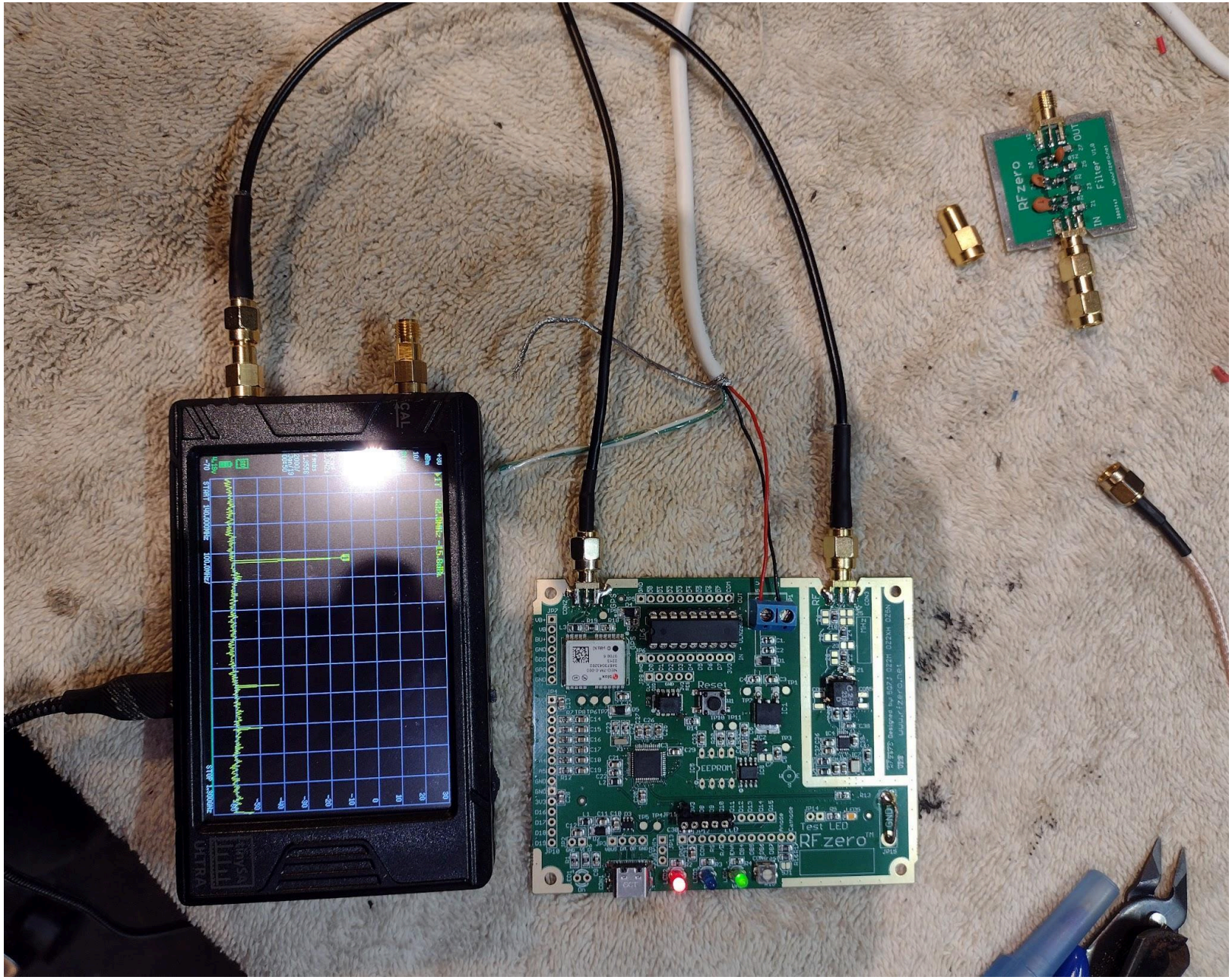
Decoding

To decode a PI4 signal you will have to set your receiver to the same frequency as you would do to receive a beacon with an 800 Hz carrier. The receiver bandwidth should not be less than 1 kHz. On most receivers you will have to set your USB dial 800 Hz lower than the nominal frequency, e.g. OZ7IGY is assigned to transmit on 50,471 MHz thus the receiver must be tuned to 50.470.200 to show a 0 Hz frequency deviation. When you receive the carrier at exactly 800 Hz then you know how accurate your station is.

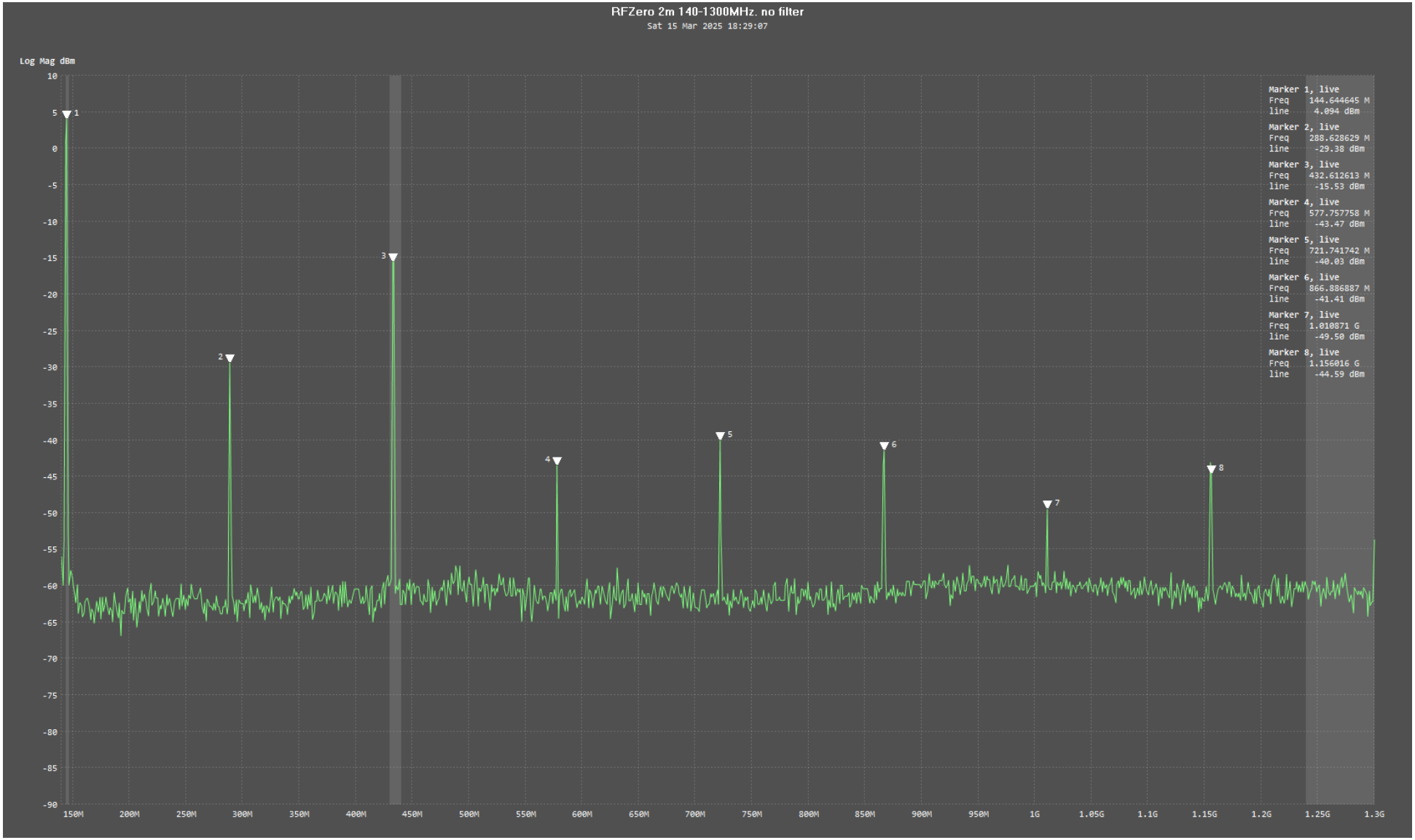
To decode PI4 you can use

- PI-RX by Poul-Erik, OZ1CKG, Windows
- MSHV by Christo, LZ2HV, Linux and Windows
- Michael, DG0OPK, running MSHV on a Cubieboard2
- SparkSDR by Alan, M0NNB Linux, Mac and Windows

The Hardware: RFZero board programmed for PI4+CW+Carrier



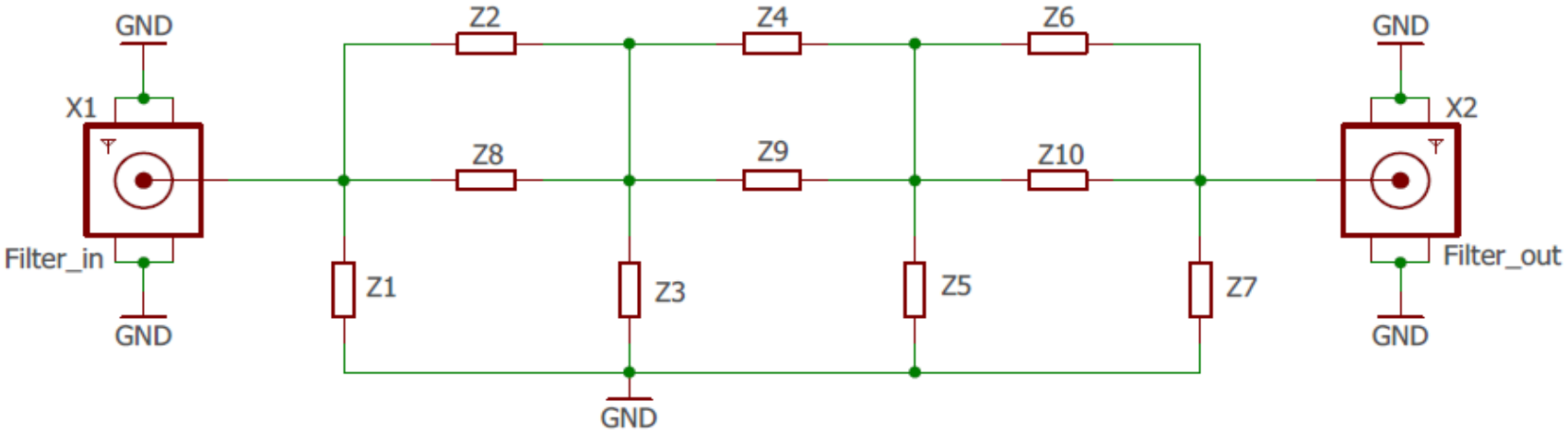
Spectrum response without filter, tinySA w/30dB attenuation



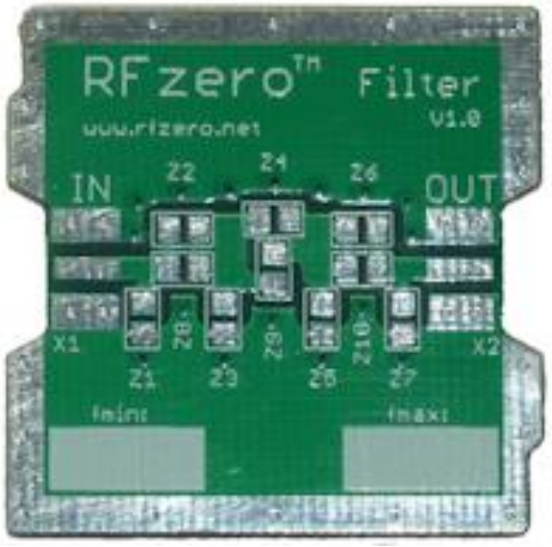
Effect of 100-150 MHz. low pass filter specified at https://rfzero.net/shop/

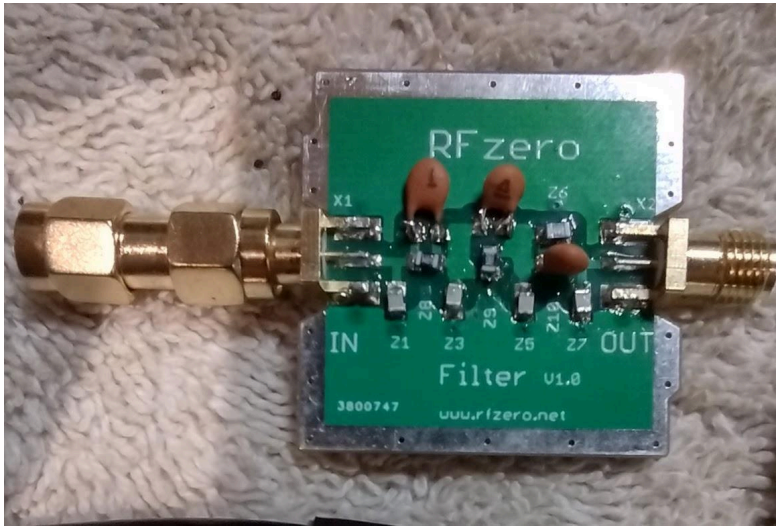
The Filter PCB has ten positions that can be used for either a low pass filter or a high pass filter. It has the same topology as the RFzero output filter itself. The generic filter topology where each Z# is replaced with either a C or L or shorted/omitted as below. The boards are designed for 0805 SMD components.

The value and type of each Z# depends on the design criteria, e.g. for a low pass filter Z1, Z3, Z5 and Z7 are capacitors and Z8, Z9 and Z10 are inductors and if Z2, Z4 and Z6 are used they are capacitors. Not all Z# have to be used. If so they can be omitted or shorted, whichever applies.



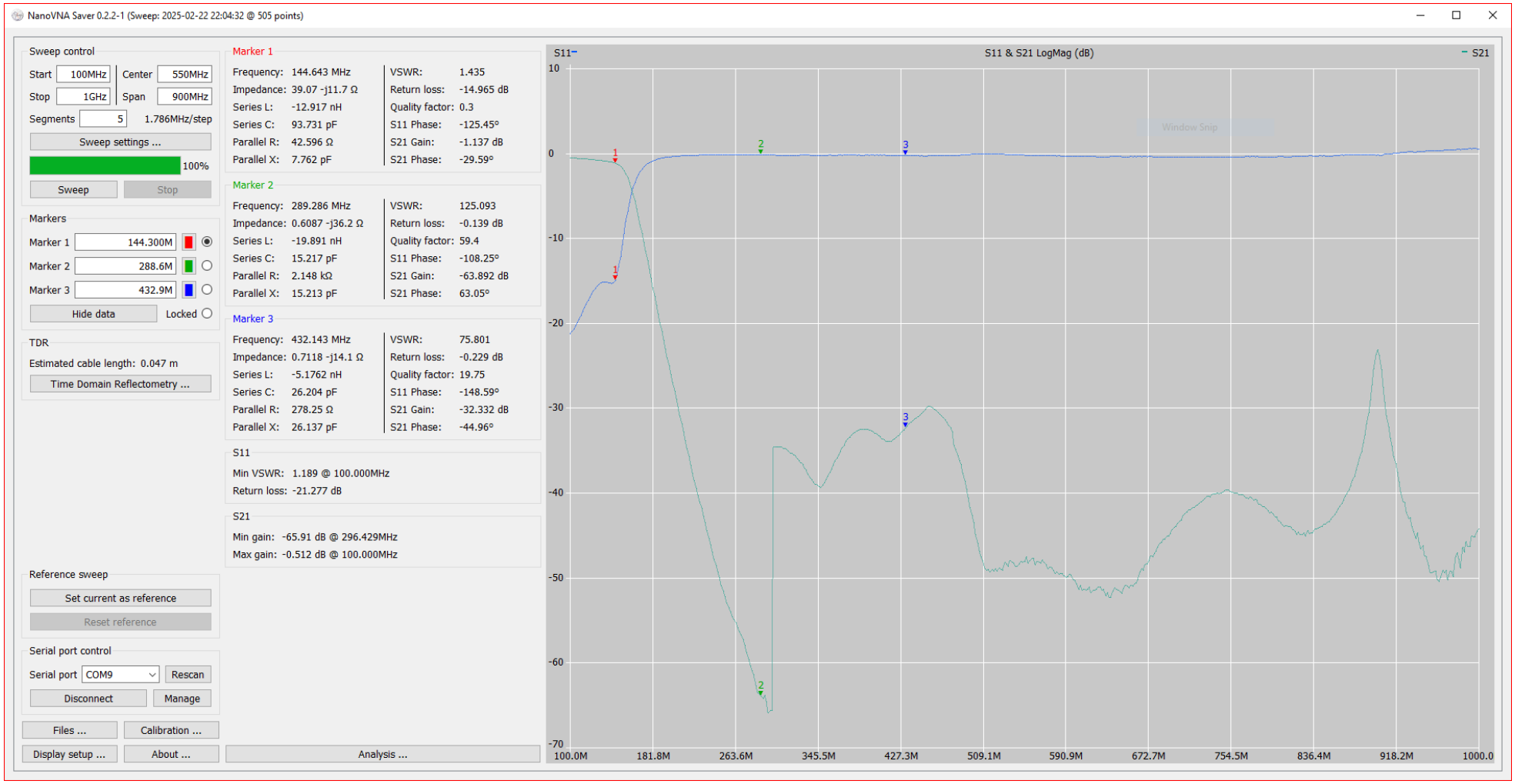
fmin	fmax	Z1	Z2	Z3	Z4	Z5	Z6	Z7	Z8	Z9	Z10
100MHz.	150MHz	15pF	1pf	33pF	3.9pF	33pF	3.3pF	12pF	68nH	68nH	68nH





nanoVNA measured filter response; this response is typical of an elliptic filter ladder:

https://www.sciencedirect.com/topics/engineering/elliptic-filters



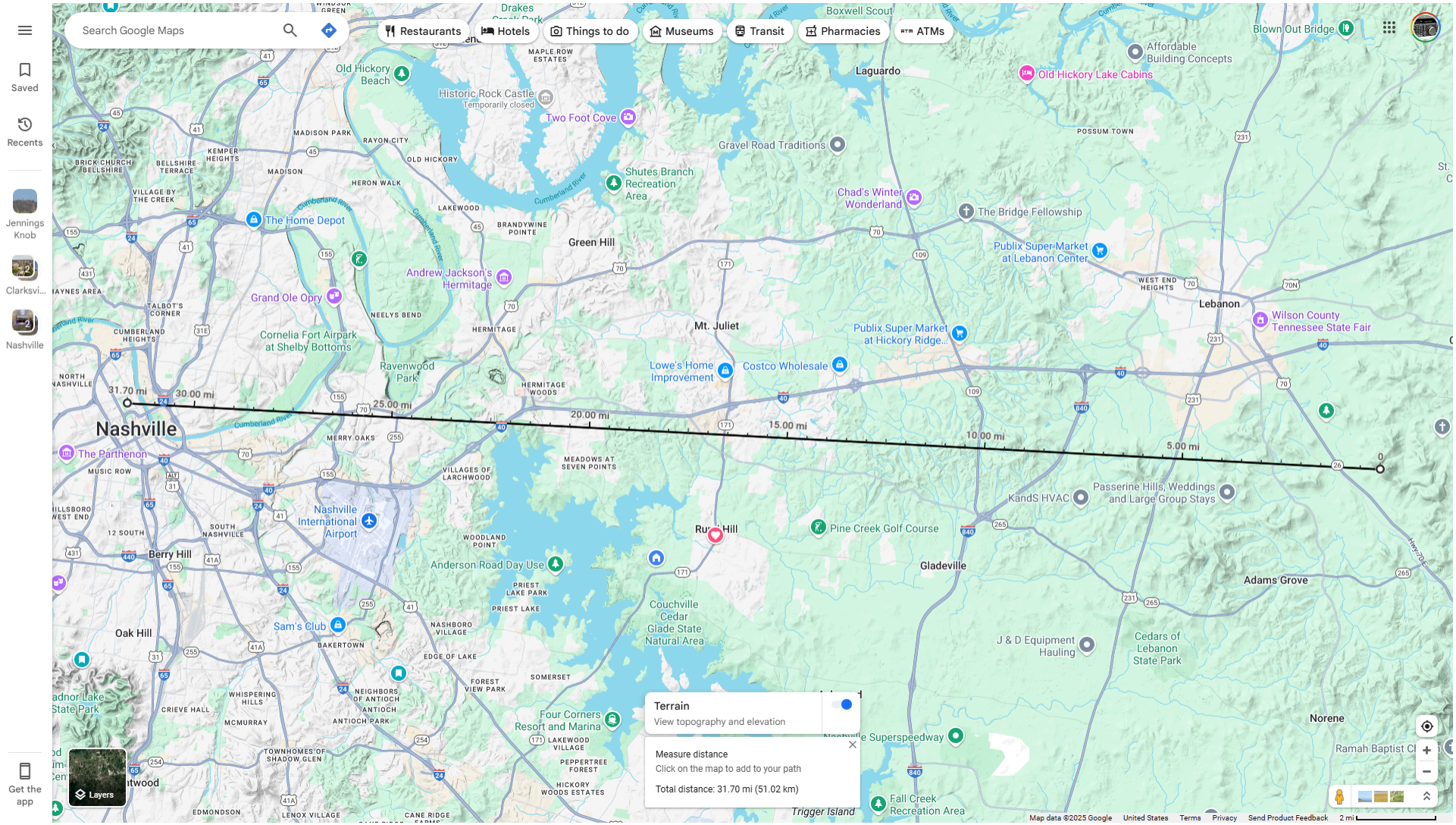
RFzero board, low-pass filter board, power amplifier, power supply, enclosure

Beacon modes; something beyond a carrier & callsign; digi vs. traditional, intro to mixed mode operation; use RFZero presentations as basis; band plan/coordination, IARU 1 vs. 2; lack of coordination in region 2

RFZero Manager; canned utilities; makes using the RFZero really easy
Arduino IDE; modifying code
Receiving PI4

Construction photos, filters, power amps, antenna
Installation photos

The beacon site is ~32 miles from Nashville. Site ground elevation is 1220’, tower is ~80’. Commercial power with generator backup is available.



Western view, Nashville hardly visible in the haze.

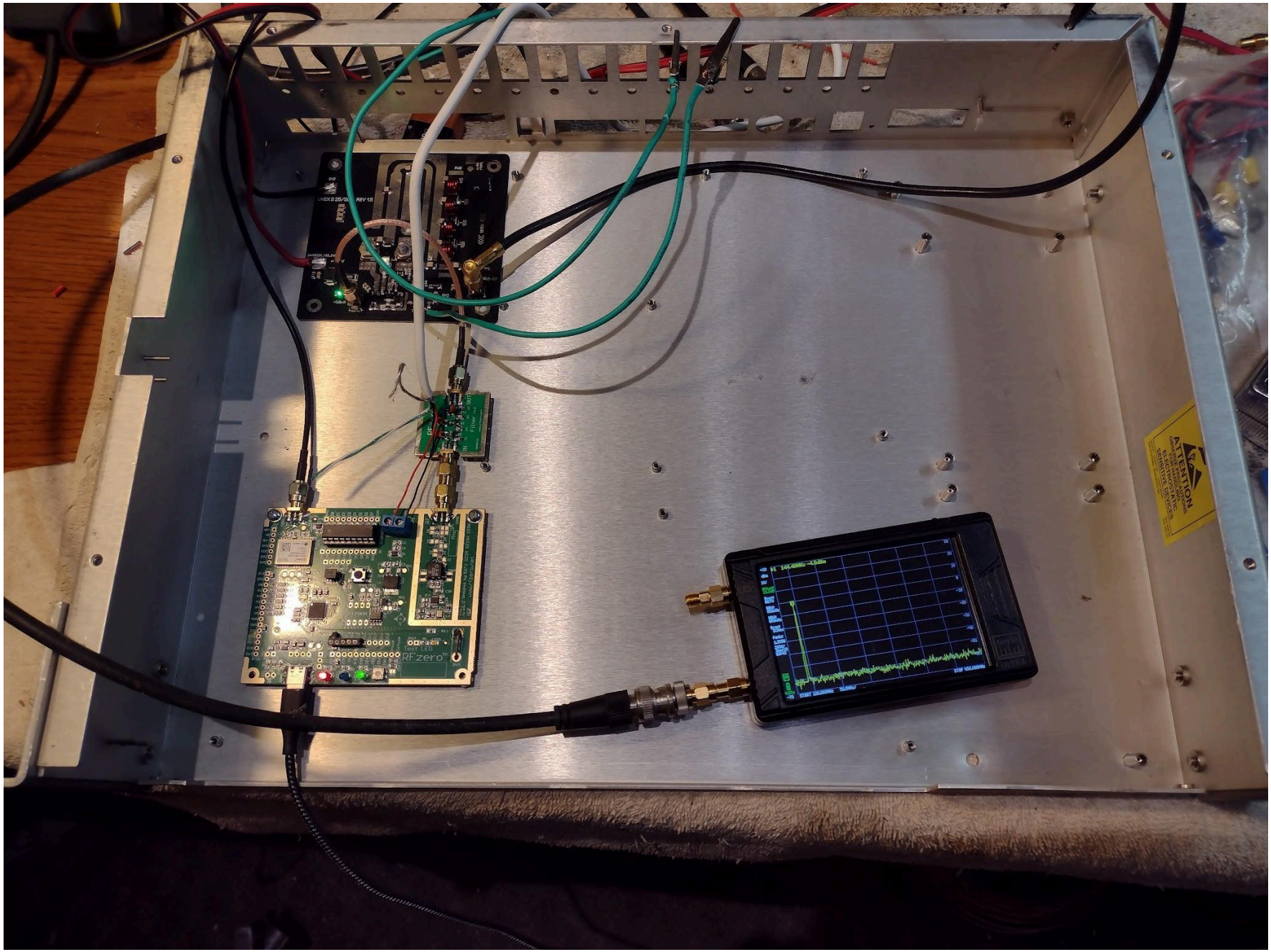
Northern view; cooling tower for the Hartsville, TN TVA Nuclear Power Plant ~20 miles away; $1.5 billion project began in 1977, never completed. Our government hardly at work…

Antenna under construction: it's based on a variation of the Big Wheel & developed by L.B. Cebik W4RNL and Bob Cerreto WA1FXT and documented in a March, 2008 QST article called "A New Spin on the Big Wheel"
http://radio-amador.net/pipermail/cluster/attachments/20100107/a2f81f3e/AntenaHPOD.pdf
It will be a stacked pair of three dipoles bent into a circle, each dipole ½ wavelength. A hollow, triangular mast with radial arms will support the ⅜" copper tubing elements. The element insulators are poplar hardwood dowels 5/16" in

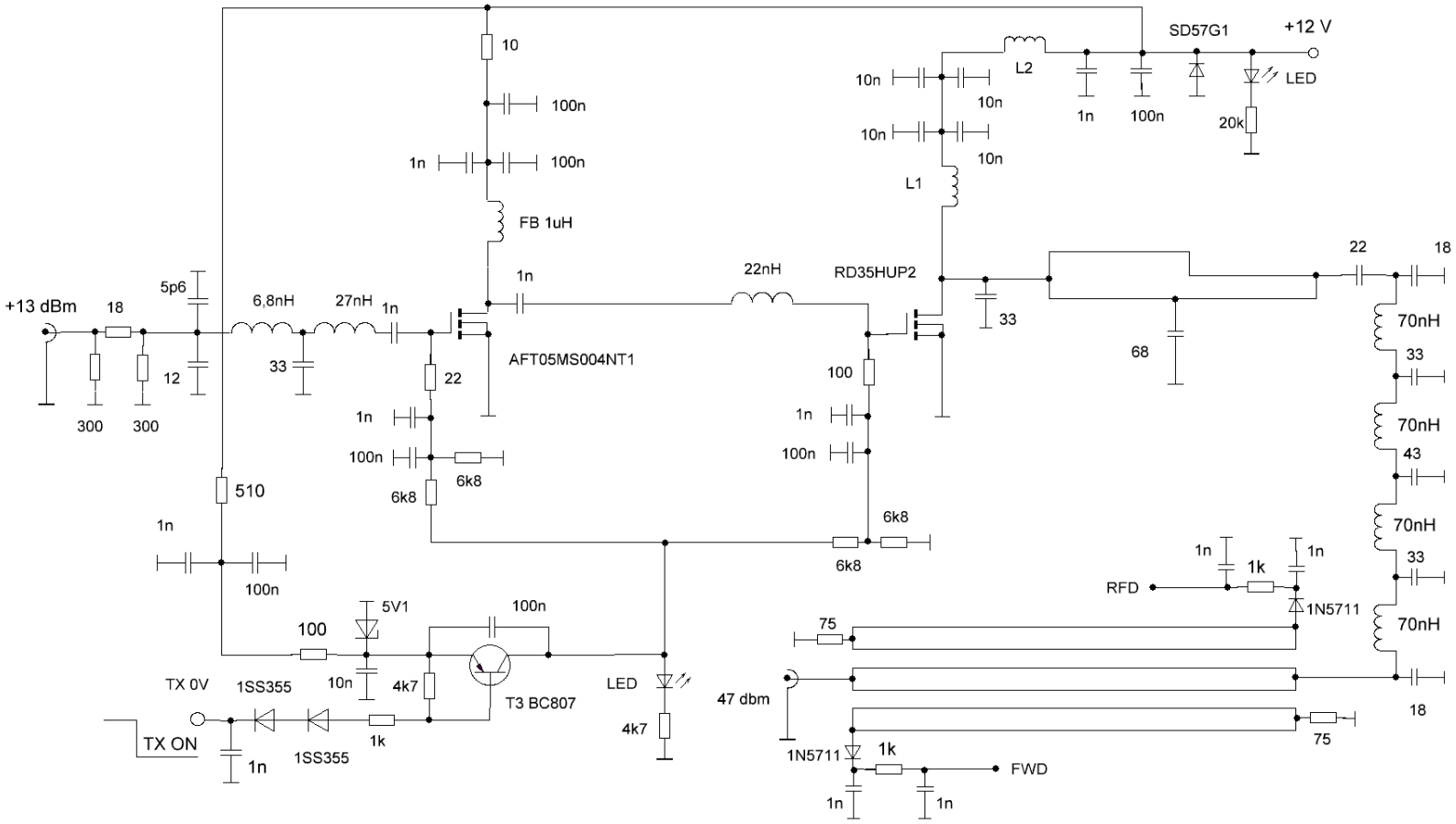
diameter (NOT the ⅜" dowels in the photo). Each set of three dipoles is connected in series, and the two pair of three connected in parallel resulting in a feedpoint impedance of ~70 ohms, which is easily matched to 50 ohms. Construction details are in the magazine article.

I’ve been experimenting with varying the tip to tip spacing of the elements in order to achieve resonance at 144.300MHz. with a resulting feedpoint impedance of ~33 ohms per dipole, which will give a final array impedance of very near 50 ohms, thereby eliminating the need for a matching network. So far, no dice..but hope springs eternal.

The chassis



The P.A. by Sergey EX8MLE
https://rfzero.net/documentation/third-party-solutions/pa-50-mhz-aft05ms031n/



Board material - FR-4, thickness 1,6 mm

L1 - PCB trace, W - 2,54 mm, L - 160 mm.

L2 - PCB trace, W - 2,54 mm, L - 65 mm.

LPF coils - 4t, ID 6 mm, L-5,5 mm, wire 1 mm

P.A. temporarily fastened to the floor of a repurposed aluminum rack mount chassis. This confirmed that a more substantial heatsink is required. Efforts continue to find the right balance of P.A. supply voltage, drive level and output power while maintaining a good thermal transfer characteristic.