John Golomb, N2NC / john.golomb@gmail.com

Transforming the K3 into a SDR + K3

My SDR Journey

This article will explain a few of the ways I use an external SDR and *Linrad* to enhance my K3 experience. *Linrad* is one of many available SDR receiver programs. The same could be done with any radio that lets you access its IF frequency output. I have been using *Linrad* as my "baseband" receiver. The K3 is used only up to its 8.215 MHz IF.

Developed by Leif, SM5BSZ, *Linrad* originally was written for Linux, but now it runs on Windows as well. It has a graphical user interface, but, given its 1990s heritage, does not look like a modern Windows program. Don't let that fool you. It is a very high-performance receiver. One unique thing about *Linrad* is it allows the user to customize the receiver for various applications. The learning curve can be steep for newcomers, but it's not that hard. Default settings are good enough to get started. Once you learn the GUI controls, the user interface has more of an "analog radio" control feel than other SDR programs. The source code is freely available, and the license allows the code to be used "by anyone for any purpose."

Alternate Receiver with Panadapter

L-P Pan (from N8LPs TelePost Inc) is a direct conversion quadrature receiver that converts the IF output of a receiver to baseband audio I and Q outputs. When coupled with customized SDR software, it provides an integrated real-time display. Eventually, I figured out I could use an LP-Pan connected to my Elecraft K3's IF Out port along with a decent sound card (I used an Asus Xonar U5) to use *Linrad* with the K3. I purchased the LP-Pan and sound card and, in a short time, I had *Linrad* up and running. One of the first things I did with the LP-Pan/*Linrad* combination was to start using it as a receiver in casual contest operations. This was not a turnkey process. I had to carefully match *Linrad*'s receive frequency to the K3 IF, and then manually adjust such parameters in *Linrad* as filter widths, RF gain, and BFO frequency. Despite the rather cumbersome setup, once I got it working, I loved the sound of the *Linrad* receiver.

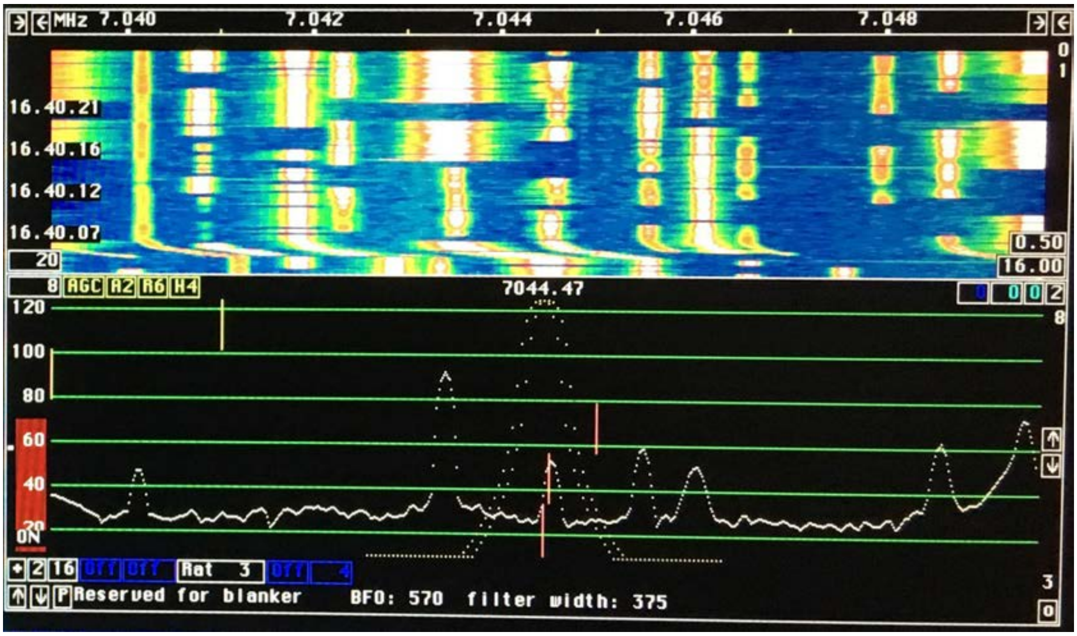
In 2018, after ARRL DX CW, Andy, N2NT (@V47T) emailed me about K3 pileup "mush" during the contest and half seriously suggested that I integrate *Linrad* into the K3. At the same time I started playing with an Airspy HF+, which is supported by *Linrad* and allowed easier setup and increased sample rate compared to the LP-Pan with external soundcard.

The *Linrad* source code has prototype "user" routines that can be used as a guide to customize *Linrad*. I am not a programmer, but with the examples and much examination of the source code, I was ready to attempt customizing the *Linrad* code to integrate it with the K3.

About 3 months before the 2018 CQ World Wide CW, I came up with a concept to poll the K3 via its serial interface for a few key K3 settings, such as mode, VFO frequency, IF (which moves around a tiny bit when you change K3 filter settings), filter bandwidth, CW pitch frequency, AGC, and RF gain and have *Linrad* with the Airspy HF+ following the K3. Since *Linrad* is actually tuned to the static 8.215 MHZ IF of the K3, such actions as using the RIT or simply spinning the "big knob" would tune the same as listening directly to the K3. Like many other SDR programs, *Linrad* provides several waterfall/spectrum displays, which I have used to make a poor man's P3.

N8LP's excellent *LP-Bridge* program was used to share the K3 serial port between the logging program and *Linrad*. I have not tested this modified code using other SDRs, but I see no reason why you couldn't use any SDR that will tune to 8.215 MHz, as long as it has adequate sensitivity.

We decided we would use my traditional ARRL November Sweepstakes CW SO2R effort from N2NT as a *Linrad* test, to decide if Andy would actually attempt using K3/Airspy/*Linrad* as his main receivers from V47T in the CQ WW CW. It worked without issue in the SS. At the end of the contest, I



Reserved for blanker BFO: 570 filter width: 375

Figure 1 — The *Linrad* baseband window. In addition to a spectrum and waterfall screen, this is the window where you control most key receiver parameters, such as RF gain, filter bandwidth and shape, BFO (CW pitch) frequency, and AGC. These track the respective K3 controls. You can also manually tweak them with a mouse.

felt that the receive sound was less fatiguing than the already very good K3. I found the *Linrad* spectrum and waterfall displays useful in the same way as I have been using the Elecraft P3/SVGA for several years. Granted, it is not quite as tightly integrated with the radio as the P3 is, but it still gives you the same situational awareness, and you can't argue with the cost saving compared to a new P3/SVGA.

Two instances of DXLog.net and two instances of *Linrad* ran on a single i5 quad core laptop with plenty of processing overhead (see Figure 1).

I'm sure that the K3/Airspy HF+ combination does not have the same dynamic range performance as the K3, but I haven't come across a case where I felt limited by dynamic range. It even handles 2-meter terrestrial VHF contests fairly well, which is probably where I encounter the strongest local signals. A multi-multi environment with two radios on the same band may be a different story.

I used *Linrad* again with WW2Y during a multiop in the 2018 SS Phone from my home station as another test. It sounds good on SSB too. In the 2019 ARRL June VHF Contest, it served as my 2 meter receiver. Weak SSB signals are easier to copy on *Linrad*, although I can't say why. One thing I like is that you can adjust the IF shift (BFO frequency) in *Linrad* in finer increments than the K3. That's a good example of *Linrad* having more of an "analog" feel and sound than the K3.

V47T did start the 2019 CQ WW CW using two *Linrad* receivers, but problems with RF getting into the setup caused Andy to abandon its use several hours into the contest.

Adaptive Diversity Reception

The K3 supports diversity reception. This requires two antennas feeding two receivers derived from common local oscillators. The traditional implementation is to use stereo headphones, with one receiver in each year. A signal will swirl between ears depending upon which antenna provides the strongest signal at any given moment. If you use a diversity receiver and configure *Linrad* to use the both receivers audio channels as input, you can take advantage of its adaptive receiver, which is controlled by the polarization graph and phasing windows seen in Figure 2. The adaptive receiver dynamically and automatically adjusts the amplitude and phase difference between the two receiver channels and combines them into a single channel that maximizes the signal in the passband. This eliminates the signal shifting between ears described earlier. I took advantage of a semi-serious multioperator effort from K3LR in the 2019 CQ 160 CW contest to evaluate the usefulness of the adaptive receiver. For most of the weekend, the main receiver used a HI-Z vertical array and the main transmit antenna (a vertical 3 element Yagi array) fed the sub receiver. This is the combination I have been using for years as input to Tim's NCC-2 during phone DX contests. Everyone that used the adaptive RX declared it a winner.

One thing I learned during the K3LR evaluation is that the adaptive function has a hard time adapting to very weak signals. To deal with this, *Linrad* gives you the option of setting the output to "X+Y" mode which feeds the "optimum" channel into the left ear and the "opposite" (reversed phase and amplitude balance from the "optimum" channel) into the right ear. When the adaptive receiver lines things up properly, you hear all of the signal in the left ear and no signal in the right ear. When there is not enough signal for the receiver to adapt to the optimum combination, you don't miss much if you listen to both X and Y channels. After the CQ WW 160 CW, K3LR and I added a Behringer XENYX 502 audio mixer, so we could run in X + Y made and tailor the balance of the X and Y audio channels to our liking. I like to center the X channel and put a little of the Y channel in my right ear (see Figure 2). It's fascinating to watch these screens update in real time. One problem I haven't fully overcome is that CW sidetone is fed into both *Linrad* receiver channels, and the receiver adapts to the sidetone. Ideally, it should not adapt to sidetone. I did modify the source code to skip the adaptive calculations if both channels are nearly identical in magnitude and phase, which helps ignore sidetone quite a bit. Several videos of the adaptive receiver in use during the CQWW 160 CW from K3LR are available on the *NCJ* website. (I apologize for the very loud and distorted sidetone in these clips; 60-Hz hum was getting into the system during transmit.

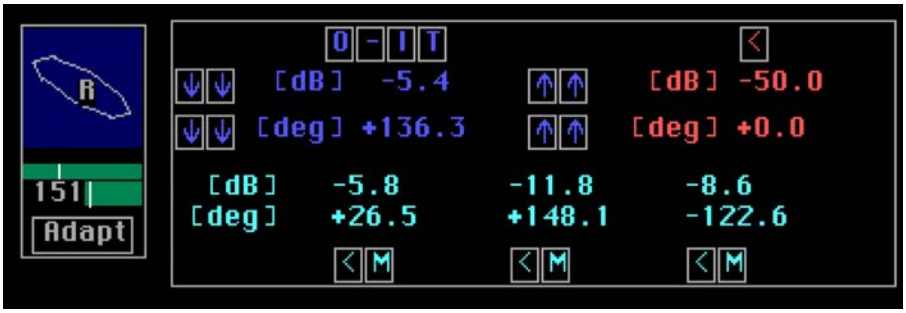
High-Performance Noise Blanker

This past winter, some pesky line noise showed up, which made me revisit the *Linrad* noise blanker. One thing we can do with SDRs is take advantage of wideband noise blanking. I am not an expert, but my simple understanding is at wider bandwidths, you cut shorter duration holes in the signal when blanking a noise impulse. The *Linrad* noise blanker has a unique architecture. A selective limiter is used to divide the wideband signal into strong signals and weak signals. Only the weak signals are routed to the blanker, which eliminates the problems of strong-signal distortion that conventional blankers produce. The weak and strong signals are then summed for processing through the rest of the *Linrad* receiver chain. The *Linrad* blanker, combined with the 768-kHz bandwidth of the Airspy HF+, has been a big plus when line noise pops up. It makes the noise a non-issue, and I notice no distortion, even on a band packed with strong signals. To learn more, visit SM5BSZs "*Linrad* For Newcomers."

You have to activate an additional high-resolution spectrum window to the *Linrad* screen to control the blanker. Activating this window also provides an optional display of oscilloscope traces of the strong and weak signals that are a very useful when adjusting the blanker. I modified the code to reduce the screen real estate taken up by the scope traces (see Figure 3). The online version of this article includes video that shows the noise blanker in action.

Digital-Mode, Fixed-Frequency Receiver

If you have operated on 6 meters during recent VHF contests, you know there's a lot of JT digital mode activity. For the 2019 ARRL January VHF Contest, I made some last-minute changes to allow *Linrad* to be set to a fixed frequency other than the frequency of the K3. I use *Virtual Audio*



[dB] -5.4
[deg] +136.3
[dB] -50.0
[deg] +0.0
[dB] -5.8 -11.8 -8.6
[deg] +26.5 +148.1 -122.6

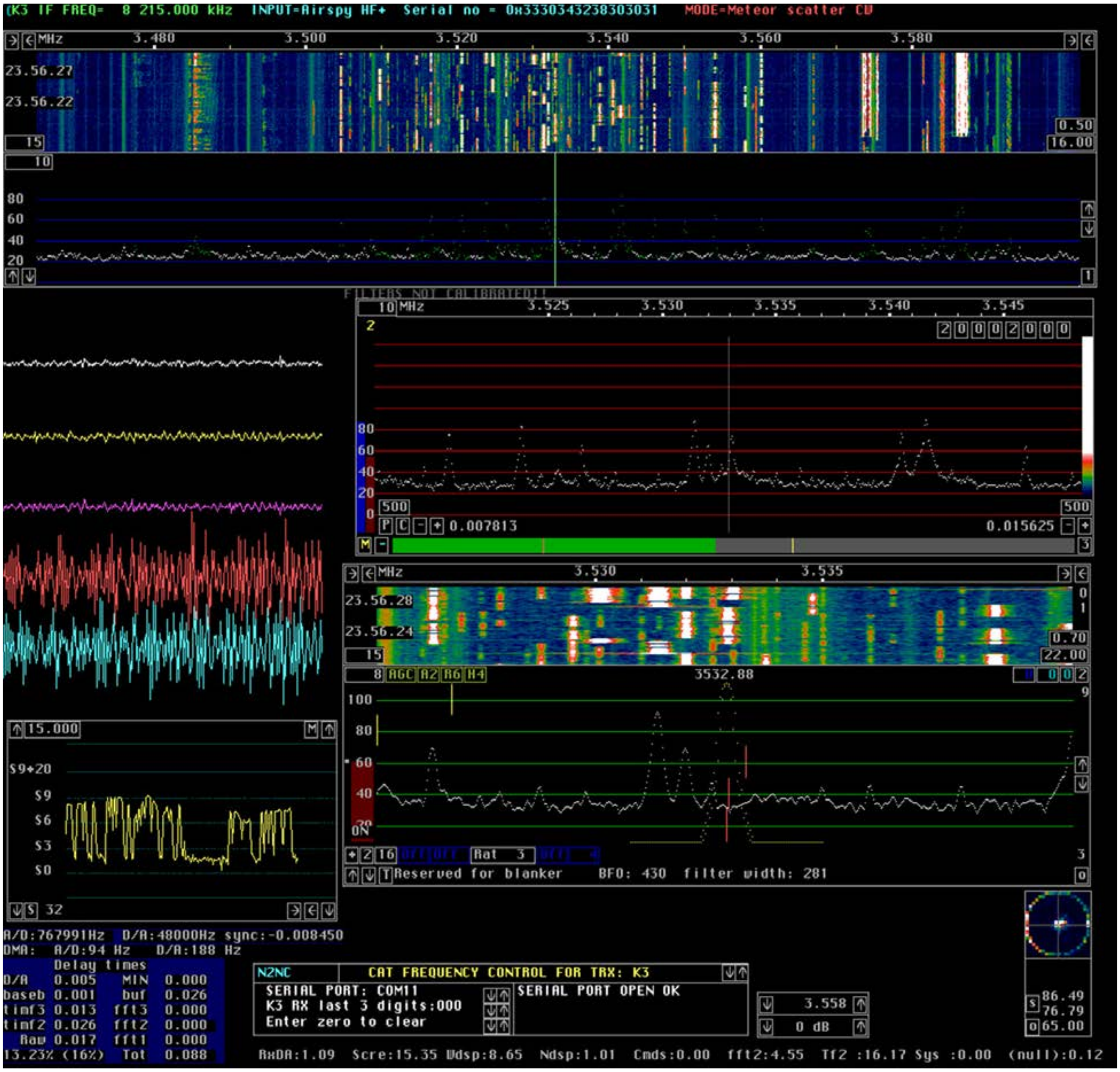
Figure 2 — The small window on the left is the *Linrad* polarization graph, which graphically displays phasing and amplitude balance. It is also used to set the time constant of the adaptive receive mode. The other window is a phasing window, an extension of the polarization graph which displays amplitude and phase information numerically. You can also use the controls in this window to manually adjust amplitude and phase.

Cable software from Eugene Muzychenko to send *Linrad* audio to *WSJT-X*. It's pretty cool to be able to call CQ on SSB or CW and keep an eye on FT8 or MSK decodes at the same time. We can also decode MSK144 while working FT8, and vice-versa. I was amazed at how many FT8 decodes we got, even with interruptions from SSB CQs. The monitor screens do get a bit busy in VHF contests with two instances of *WSJT-X* running. We also find that we would often get MSK decodes on *Linrad* and not on the K3, or vice versa. I believe FlexRadios and some other SDRs can do this, but here is a way to add similar capability to the K3 with a $200 external SDR. The Airspy HF+ Discovery has a price below $200.

Dealing with Latency

One unavoidable issue is additional receiver latency compared to that of a stand-alone K3. If you press the "T" key in *Linrad*, it will display timing information for each stage of the receiver and the sum of all latency. I've found that if I keep total latency below about 100 ms, it is barely noticeable and does not slow down the pace in a contest, even in the NA CW Sprint. If you listen to *Linrad* audio while you transmit, you quickly learn that, due to the latency, you can't *manually* send CW using *Linrad* as your sidetone, and, believe it or not, it is hard to talk if your SSB monitor has latency as well.

To solve that problem, I built a simple circuit that uses reed relays to switch to the K3 audio when the PTT is activated (see Figure 4).


```
(K3 IF FREQ=  8 215.000 kHz  INPUT=Airspy HF+  Serial no = 0x3330343238303031   MODE=Meteor scatter CW
```

```
Reserved for blanker     BFO: 430   filter width: 281
A/D:767991Hz  D/A:48000Hz sync:-0.008450
DMA:   A/D:94 Hz    D/A:188 Hz
       Delay times
D/A    0.005   MIN   0.000
baseb  0.001   buf   0.026
timf3  0.013   fft3  0.000
timf2  0.026   fft2  0.000
  Raw  0.017   fft1  0.000
13.23% (16%)   Tot   0.088
CAT FREQUENCY CONTROL FOR TRX: K3
SERIAL PORT: COM11                SERIAL PORT OPEN OK
K3 RX last 3 digits:000
Enter zero to clear
3.558
0 dB
86.49
76.79
65.00
RxDA:1.09  Scre:15.35 Wdsp:8.65  Ndsp:1.01  Cmds:0.00  fft2:4.55  Tf2 :16.17 Sys :0.00  (null):0.12
```

Figure 3 — My typical screen layout during contests. At the top is a wide waterfall/spectrum display. The next window down is the high-resolution spectrum, which has the noise blanker controls. The oscilloscope traces let you view the "weak" signals that the blanker acts upon (top 3 traces) and the "strong" signals (bottom 2 signals). Below the high-resolution spectrum is the baseband window and waterfall. The bottom left of the screen shows information on delay times through different components of the receiver. The remaining windows at the bottom of the screen are related to the radio interface.

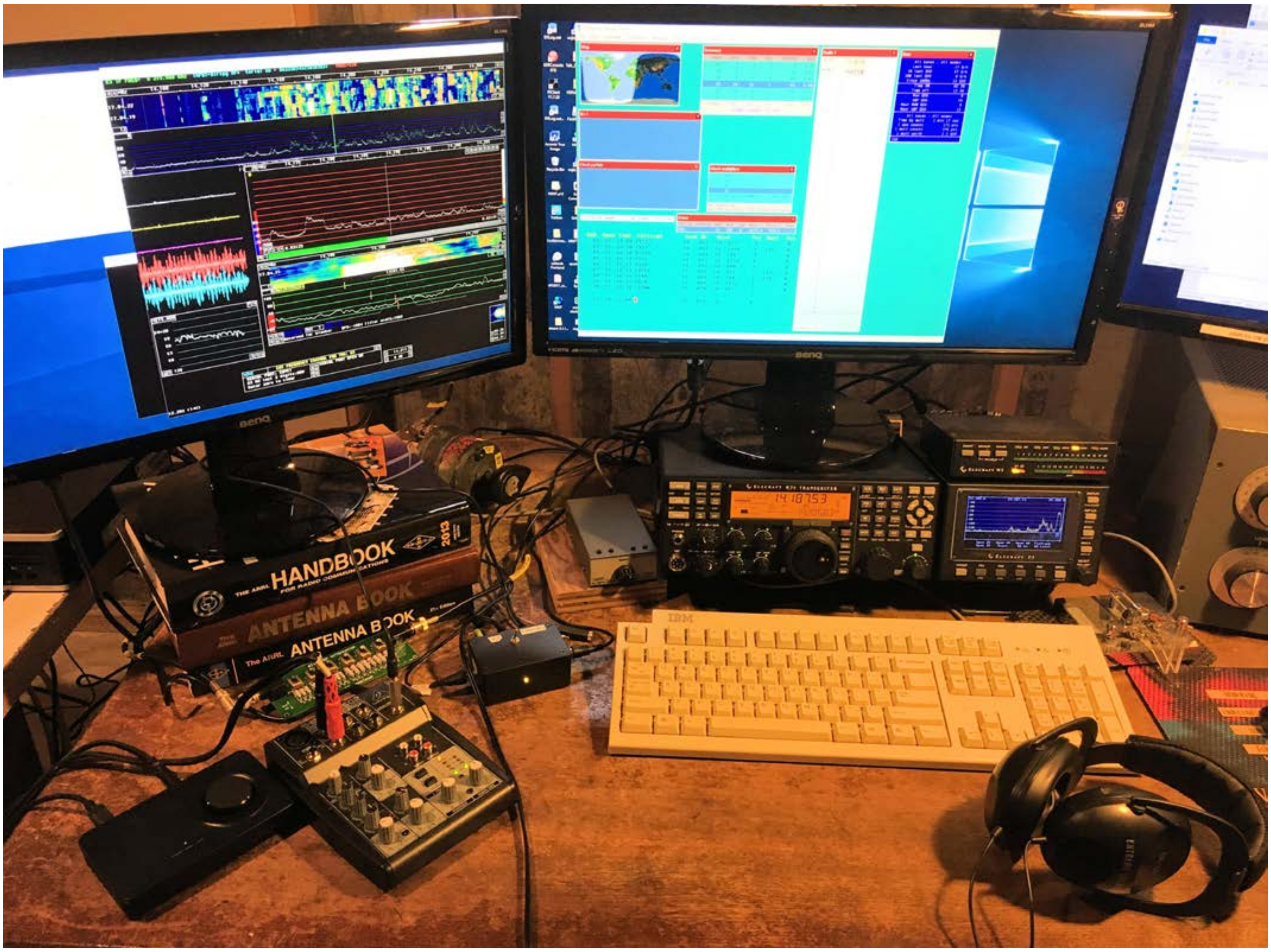


Figure 4 — My station running *Linrad* during the 2019 WPX SSB. The three boxes specific to using *Linrad* (left to right) are the Asus Xonar USB sound card, used as *Linrad* audio output; the Xenyx 502 mixer console, and the *Linrad*/K3 audio switchbox. Not shown is the very small Airspy HF+ SDR, plugged into the IF OUT port of the K3.

Final Hints

I have placed my modified *Linrad* executable and setup information on my Google drive for anyone interested; the URL is available in the online version of this article on the *NCJ* website, **http://ncjweb.com**.

Here are some hints I have learned over the past few years while using *Linrad*.

Use a step attenuator at the input of the HF+ to set the noise level similar as you switch from band to band. Don't waste dynamic range on excess noise. You can also keep the waterfalls "biased" correctly by adjusting the attenuator, instead of changing *Linrad* parameters.

Sound card drivers affect latency. Using a sound card that supports the ASIO protocol can help with latency. Also, make sure the output sampling rate set in *Linrad* matches the sampling rate set using the Windows sound control panel.

Check that your anti-virus software is not monitoring the *Linrad* executable. I have had that happen a few times and resolved the issue by manually adding *Linrad* to the exclusion list.

Press T to in *Linrad* to monitor the total latency and processor usage in the lower left-hand corner of the screen. On my PC, *Linrad* input and output synchronization can get disturbed by other running programs. This will manifest itself as a clicks in the receive audio and may also add latency. This usually is caused by web-browser operation, not logging software. It may be due to disk access. A solid state drive may help. The *Linrad* computer during the K3LR tests had a solid-state drive, and total latency was very stable. I generally keep web browsers closed when running *Linrad* in a contest.

Another timing issue is sending CW. I have always used com port RTS/DTR for CW and PTT control from logging programs. CW can get choppy if it uses the same port that *LP-Bridge* shares between logging software and *Linrad*. If you want to ensure no CW timing problems with RTS/DTR keying, use a separate, dedicated COM port that is not shared with *LP-Bridge*. This also means using the traditional **KEY** and **PTT IN** jacks on the K3. I have never used Winkey, so I can't comment on its use.

Extensive help files are built into *Linrad*. If you want to know what a screen control does, place the mouse over the control and hit the F1 key. Another great way to learn *Linrad* setup and use is to watch the many videos Leif has posted on his YouTube page.

Thanks to SM5BSZ, ON5GN, K1JT, EA4GPZ, W3SZ, and others who wrote the code examples included with the *Linrad* source files. They greatly helped me during this project.