Linrad: *New Possibilities for the Communications Experimenter, Part 4*

Examples of simple Linrad *use with an amateur transceiver (IC-706) yield improved noise blanking and filtering*

By Leif Åsbrink, SM5BSZ

Linrad is designed to fit any hardware that can supply a digital data stream to a PC computer. Of course, station performance improves with two channels having a large bandwidth, but running *Linrad* with only one channel at a bandwidth of a few kilohertz can produce dramatic performance improvements for a conventional amateur receiver. *Linrad* is still at an early stage of development and the only processing mode currently available is "weak-signal CW." This mode has no AGC and it is fine for 144-MHz EME. It can be used for ordinary CW and SSB just by setting suitable bandwidths, but dedicated modes with AGC and AFC optimized for these activities will be incorporated in the future.

Dynamic Range with and without Noise Blanker

Dynamic range, the ability of a receiver to receive a weak signal without degradation in the presence of one or more strong signals, is one of the most important characteristics of a receiver. It is well known that intermodulation-distortion measurements are often inaccurate. See, for example, Ulrich Rohde's article in *QEX*, Jan/Feb 2003. Also the dynamic range achievable when there is only one strong undesired signal is often seriously incorrect due to inadequate measurement procedures.

Before going into how *Linrad* can be used to improve the dynamic range of a receiver, I think it is appropriate to define things carefully. Firstly, definitions may differ in a way that makes numbers very different while the actual measurement behind the numbers is the same. Such measurements can be converted from one definition to another. Secondly, completely different dynamic-range properties of a receiver may be characterized by similar names. If such measurements are compared with each other, the result is meaningless.

All dynamic-range measurements relate something to the weakest signal that can be received. Let us call it MDS (for minimum discernible signal) without going into details at this point.

Dynamic-range measurements can be classified into four types:

1. *One-signal dynamic range:* The power of the strongest signal that can be received without distortion divided by the power of the MDS.

2. *Two-signal dynamic range:* The power of the strongest signal that can be present at the same time as a signal at the MDS power level is received, divided by the MDS power level.

3. *Three-signal dynamic range:* The power of the strongest signal pair that can be present at the same time as a signal at the MDS power level is received, divided by the MDS power level. Both interfering signals have the same power level and the power to divide by the MDS power is the power of one of them.

Type 1 is not often interesting in Amateur Radio. Very good values can be achieved by use of AGC in RF and IF sections. The ARRL lab has their own definition of blocking dynamic range which is denoted BDR* in this article. BDR* measurements of the FT-1000D show that this receiver can tolerate a 0.1 W signal into the front end without blocking.

Type 2 is typically limited by reciprocal mixing due to phase noise in the local oscillator(s). This dynamic range can be greatly improved by *Linrad*

Jaders Prastgard 3265
63505 Eskilstuna, Sweden
leif.asbrink@mbox300.swipnet.se

under certain circumstances, as will be shown below. State-of-the-art variable-frequency local oscillators set a limit somewhere around 140 dBc/Hz with 20-kHz frequency separation, which corresponds to 116 dB in 250 Hz. BDR* values are often much higher and should not be confused with type-2 measurements.

Type 3 has been discussed extensively in amateur literature. Details are beyond the scope of this article, but the setup described below is easily extended to a very accurate and reproducible method for measuring intermodulation-free dynamic range.

To make the above definitions exact, one must define the MDS and a precise level of degradation as negligible. MDS can be defined in many ways and it may depend on the mode CW, SSB or FM. For the purpose of comparing amateur receivers—where the interesting dynamic range limitations, type 2 and type 3, originate in circuits ahead of the bandwidth limiting filter—it is enough to measure MDS in one mode. It is natural then to use one of the linear modes CW or SSB.

The ultimate limit for receiver sensitivity is the noise floor. The MDS is directly proportional to the noise floor, and I propose using the power level of the noise floor in 1 Hz bandwidth to define the MDS power. For the definitions, I assume the receiver is connected to a signal source that is impedance-matched to the nominal impedance of the receiver, typically 50 Ω. A completely noise-free receiver then has the MDS power equal to k*T*, where k is Boltzmann's constant (1.38066×10^{-23}) and *T* is the absolute temperature of the resistor. With *T* = 293 K (room temperature), one finds $P = 4.045 \times 10^{-21}$ W = –173.93 dBm. For simplicity, we define room temperature as the temperature where a resistor delivers –174 dBm for each hertz of bandwidth.

Rather than specifying the MDS power in dBm, or more precisely in dBm/Hz (it is a power density in W/Hz), one can specify the MDS power density in decibels above the room temperature resistor at –174 dBm/Hz. This number is the noise figure of the receiver and it is a generally accepted figure of merit for receiver sensitivity.

To make measurements easy and compatible to transmitter measurements, it is a good idea to define acceptable degradation as 3 dB. If the sideband noise of a transmitter with a poor local oscillator is measured to a certain power density in –dBc/Hz, an otherwise perfect receiver with the same local oscillator will have exactly the same type-2 dynamic range. This could also be specified as an effective noise floor in dBc/Hz where the "c" stands for the power of the carrier of the undesired signal.

The two-signal dynamic-range noise floor in dBc/Hz is equal to *NF*–*P*–174, where *P* is the power level (dBm) of a signal that degrades the S/N of a weak signal by 3 dB, and *NF* is the noise figure in decibels. This is a very precise definition and it is easy to set up a measurement that will give reproducible results with a known accuracy. If we want the dynamic range itself, not its associated noise floor, we must divide the other way around: *DR* = (174 + *P* – *NF*) dBc × Hz.

It is obvious what this definition means if the receiver dynamic range is limited by reciprocal mixing—and this is usually the case in modern receivers. If the dynamic range were limited by blocking, it would be less clear. There is a chance that both signal and noise are reduced simultaneously when a strong, off channel signal is applied. This kind of blocking, to the extent that it does not change S/N, can be compensated by a fast AGC and is less serious than a loss of S/N, which cannot be compensated. Blocking can be measured separately and the corresponding figure of merit is BDR*, the blocking dynamic range. It's good practice to specify BDR* in those rare cases where it's less than the two-signal dynamic range defined above.

To show how *Linrad* can be used to improve the dynamic range of an IC-706MKIIG, the setup of Fig 1 was used.

The HP-8657A generator was set to –106 dBm. The star connector contains 25-Ω resistors so the power level reaching the spectrum analyzer and the IC-706 is –130 dBm. The spectrum analyzer is connected to monitor the frequency and amplitude of the free-running vacuum-tube signal generator, which has very low phase noise but lacks calibration in both frequency and amplitude. The 10-dB attenuators are identical to within 0.1 dB, so the input signals to the IC-706 are identical to those to the HP-8591A to within 0.2 dB. Each signal level is probably accurate to within 1 dB.

When only the HP-8657 was running, the signal level as measured by the *Linrad* S-meter was 48 dB, while the noise floor was 31 dB in a 100-Hz bandwidth. A –130-dBm input signal thus produces 37 dBc/Hz, so the noise floor is at –167 dBm/Hz. That means the noise figure of the IC-706MKIIG is 7 dB.

In the measurement setup, the strong signal is passed through a filter with a deep notch at 10.683 MHz, as can seen in Fig 1. The frequency response of this filter is shown in Fig 2 and the schematic diagram is in Fig 3.

A notch filter like this is reasonably easy to design. To make it useful for intermodulation measurements, the input and output transformers are wound on rather large iron-powder cores, T80-6 from Amidon. With a high transformation ratio, the notch becomes deeper and wider. The filter I have used is rather narrow with a transformation ratio of only 4:1. The notch is then only about 5 kHz wide at the 1-dB points while the attenuation is about 60 dB at the notch center. The filter is flat from 9 to 12 MHz.

The desired signal is placed at the frequency of the notch and the sideband noise of the strong signal is improved by 40 dB by the notch filter. This way, the measurement setup has much greater dynamic range than the IC-706 or any other receiver. The amplitude of the strong signal is adjusted until the S/N for the desired signal is degraded

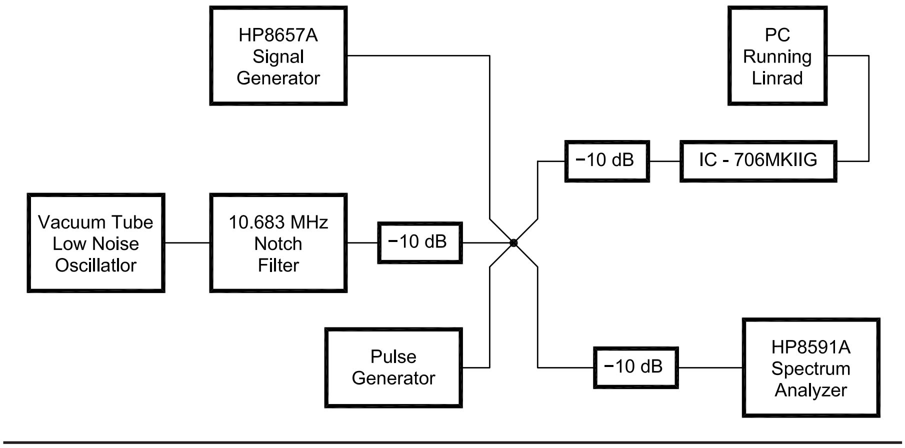


Fig 1—Setup for dynamic range measurement. One strong and one weak signal are summed and a pulse generator can be switched on to measure how the noise blanker is disturbed by strong signals.

by 3 dB, and the corresponding power level is recorded for each frequency.

The measurement procedure mimics the real situation for which the two-signal dynamic range is the relevant figure of merit. It is up to the tester to find a combination of gain control, AGC, blanker and other settings that maximize the dynamic range.

Two series of measurements were made on the IC-706: one with the weak and the strong signal only, another with the pulse generator added. For the measurement, the pulse generator was set to emit pulses repeating at 250 Hz. Without a noise blanker, the pulses lifted the noise floor of the IC-706 by 35 dB. When the IC-706 noise blanker is switched on, the noise from the pulse generator is reduced by 28 dB for a remaining degradation of 7 dB. Table 1 shows what signal level is required to degrade S/N by 3 dB in the two cases. Since the noise floor is at –167 dBm/Hz, the dynamic range is 127 dBc Hz at a frequency separation of 20 kHz.

Now, what has all this to do with *Linrad*? The interesting thing is that if the IC-706 noise blanker is switched off and the *Linrad* noise blanker is switched on, the big losses of dynamic range caused by the noise blanker disappear completely. This is a qualitative difference, not just a small improvement. The reason is obvious: The narrow filter of the IC-706 prevents the strong signal from reaching *Linrad* at all. Actually, it is even better than that because the dynamic range of the IC-706 is limited by reciprocal mixing. The strong signal can be made another 6 dB stronger before reciprocal mixing causes degradation of the elevated noise floor. With the IC-706 blanker switched off and the *Linrad* blanker switched on, the pulse generator lifts the noise floor by 6 dB, which adds one more decibel in favor of the *Linrad* blanker.

If the pulse generator were run at 500 Hz, the IC-706 blanker would not work well any more. The S/N of the desired signal is degraded by 17 dB. If the IC-706 blanker was switched off and *Linrad* allowed to take care of the problem, the 500-Hz pulse repetition frequency would cause a S/N loss of 8 dB only.

If the pulse amplitude were reduced by 6 dB for a S/N degradation of 29 dB without any noise blanker, the IC-706 blanker would reduce the degradation to 6 dB; while the *Linrad* blanker would reduce the S/N loss to below 0.5 dB at a pulse repetition frequency of 250 Hz. As soon as the pulses are small enough to stay within the linear range of the IC-706's IF, product detector and audio sections, the blanker works with mathematical precision and removes interference pulses almost completely.

In case the pulse amplitude were reduced by 20 dB, the degradation due to 500-Hz pulses would be 17 dB if no blanker were used. The IC-706 blanker does not work at all in this situation, but the *Linrad* blanker is very successful: The S/N loss is less than 3 dB. Pulses that are too weak to trigger the IC-706 blanker are nearly eliminated by the *Linrad* blanker. The 3 dB S/N loss is due to loss of signal. If the level of the desired signal were reduced to a typical 144-MHz EME level corresponding to –150 dBm into the IC-706, the signal loss caused by the blanker would affect the noise to the same degree as it affected the desired signal. The S/N loss would be a few tenths of a decibel only. The pulses have to be made much stronger even at 500-Hz repetition rate to give any noticeable S/N degradation.

I do not know how the IC-706MKIIG compares to other receivers, but I believe the results for the IC-706 are typical. What may differ is the linearity and noise floor of the IF, product detector and audio section. A traditional noise blanker is always a compromise. More bandwidth for the blanker improves pulse suppression and allows somewhat stronger undesired signals in the blanker passband. On the other hand, the number of strong signals and the risk for getting a very strong signal into the passband increases with blanker bandwidth. A variable blanker threshold is valuable; if there are no strong signals to worry about one can set the threshold low and eliminate weaker pulses.

How the *Linrad* Noise Blanker Works

Unlike conventional blankers, the *Linrad* noise blanker does not gate out the entire signal for the duration of a noise spike. *Linrad* has a calibration procedure during which a pulse generator is used to send pulses into the antenna connector of the radio hardware. The calibration pulses are typically 20 ns and they have a flat spectrum from dc to 30 MHz. *Linrad* assumes the radio hardware is perfectly linear and can calculate the exact properties of the entire filter chain that is between the antenna and the digital world inside the PC. Knowing exactly what total filter response the signal has been subjected to, *Linrad* can add one more filter in the signal path that gives the total filter chain any desired characteristics that are compatible with modest gain in the digital filter. It is of course impossible to recover frequencies that have been strongly attenuated without serious loss of dynamic range.

Thanks to the calibration, *Linrad* has an optimum pulse response for the available bandwidth. This makes it easier to locate pulses. *Linrad* also knows the exact shape of an interference pulse, so it will subtract the known shape from the data stream. The data stream on which the blanker operates does not contain any strong signals. The first FFT is used to split the incoming signal in two groups. One group contains all strong signals, the other contains all weak signals and the noise floor.

Since most of the spectrum belongs to the weak group, most of the pulse energy is there and the pulses are not much distorted. To compensate for the distortion, one just divides the peak amplitude by the square root of the fraction that the weak signals constitute out of the entire spectrum. This way, the correct pulse is subtracted. Pulses are correctly subtracted from the entire signal—the strong signals too, despite the fact that they were excluded from the blanker input data. At present, the need of using MMX instructions destroys the blanker operation on strong signals. There are several other complications arising from the fact that *Linrad* is designed to work with two channels at 96 kHz bandwidth on a 600 MHz Pentium III. By the time computers are fast enough, it will probably be more interesting to increase the bandwidth than to avoid the CPU-load-related complications that are already in the code; but someday, they can be removed, which will make setup somewhat easier.

Table 1—Level of Interference Required to Degrade Sensitivity by 3 dB at Different Frequency Separations

Offset (kHz)	*Pulses Off (dBm)*	*Pulses On (dBm)*	*Loss due to Pulses (dB)*
5	–55	–97	42
10	–47	–91	44
15	–43	–71	28
20	–40	–60	20
25	–39	–54	15

Linrad uses several averaged power spectra to decide whether a frequency bin should be routed to the group of strong signals or whether it should be routed to the blanker input. The two averaging numbers for fft1 as well as the fft2 average number therefore affect the blanker operation. There are two level controls for the blanker: One selects what S/N should be considered a strong signal. The other sets the blanker threshold that controls what signal-level peak is considered to be a noise pulse. *Linrad* also has a conventional blanker, but it is of lesser use when the input bandwidth is only 3 kHz. For more information about the *Linrad* noise blanker, look at **antennspecialisten.se/~sm5bsz/linuxdsp/blanker/leonids.htm**.

The *Linrad* Blanker on 7 MHz With the IC-706

Fig 4 shows a sequence recorded from 7 MHz. About a dozen CW stations are visible during the 35 seconds of the recording. The numbers at the left side of the waterfall show minutes and seconds of the recording. The recording was made with a pulse generator connected in parallel with the antenna. The pulse repetition frequency was set to 100 Hz and the noise floor was lifted by 30 dB when no blanker was running. The IC-706 blanker reduced the degradation caused by the pulse generator to 10 dB while the *Linrad* blanker reduced

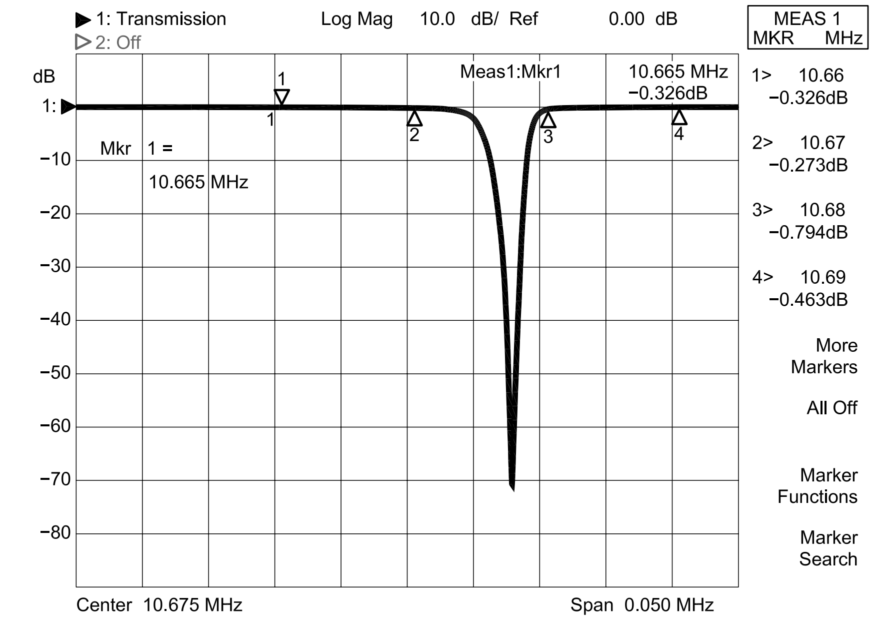


Fig 2—Frequency response of the notch filter.

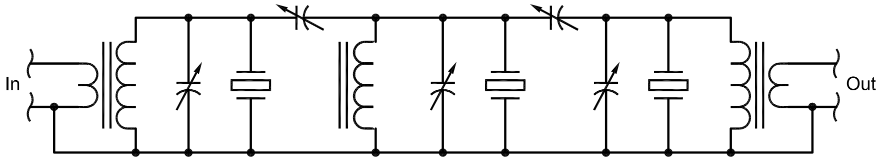


Fig 3—Schematic diagram of the notch filter.

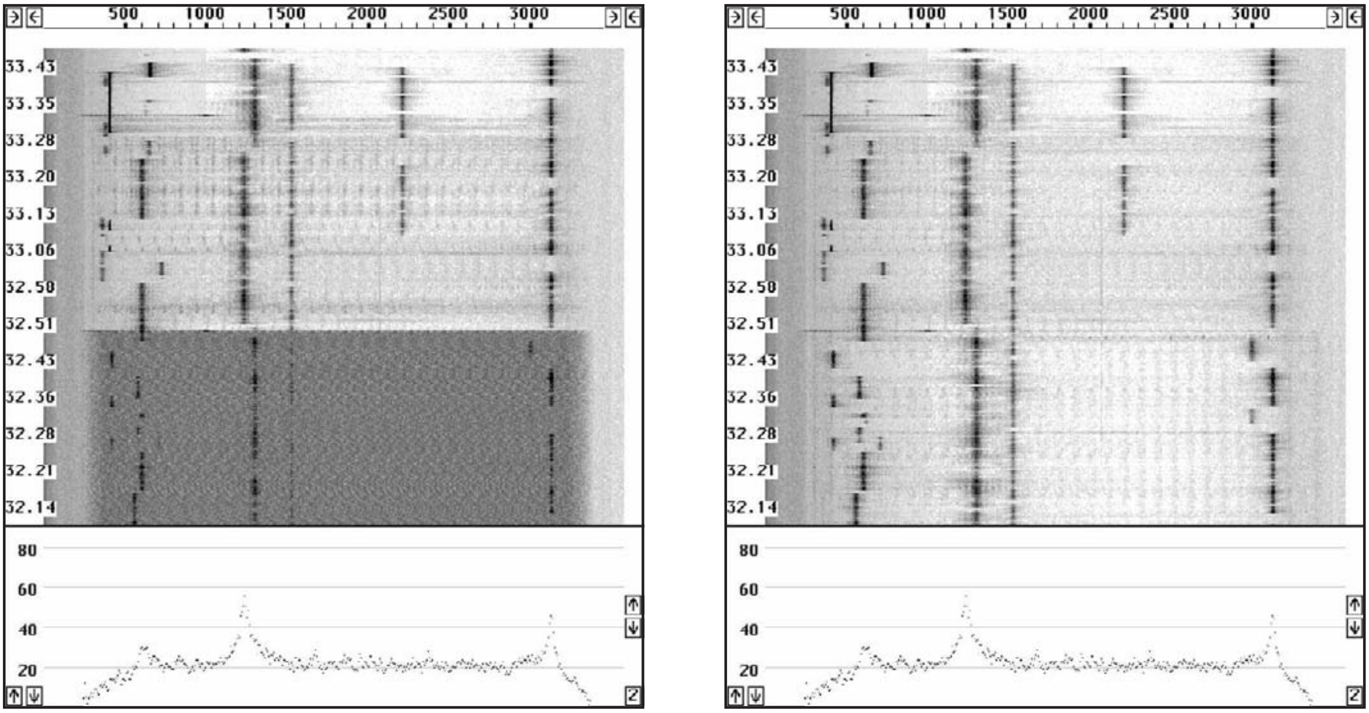


Fig 4—A digital sequence recorded with the IC-706 operating on 7 MHz processed with *Linrad*. A is with the *Linrad* noise blanker off, while the B is the same data with the *Linrad* blanker on. The pulse generator is added to the antenna signal from 32.10 to 33.30 and the IC-706 blanker is running from 32.50 to 33.32. From white to black is 40 dB. Note that the lowest noise floor is with the IC-706 blanker disabled.

it to 5 dB. The IC-706 RF gain was reduced until the AGC no longer reacted on the strongest signals and the AF volume was set just below the level where audio overtones were produced. The passband ranges from 300 to 3300 Hz and the *Linrad* mode-dependent parameters are set as described below.

Fig 4 shows the minimum improvement by use of *Linrad* instead of the built-in blanker of the IC-706 under circumstances when there is no signal within the 3 kHz passband that drives the product detector or audio section non-linear.

If the amplitude of the pulses were reduced, the difference between the built-in blanker and *Linrad* would become bigger; and if a strong signal occurred outside the 3 kHz passband, the difference would be enormous.

Setting up *Linrad* for Use With an IC-706 or Similar Receiver

A suitable sampling speed is 8 kHz. Place the 3 kHz passband of the IC-706 from 0.5 to 3.5 kHz or so. Set these parameters on the first mode parameter screen:

- First FFT bandwidth [10]
- First FFT window (power of sin) [3]
- First forward FFT version [2]
- First FFT storage time (s) [0]
- First FFT amplitude [30]
- Enable second FFT [1]

With a desired bandwidth of 10 Hz and a sine^3 window, *Linrad* selects a transform size of 1024 at a sampling speed of 8 kHz, which leads to a delay of 0.26 seconds because the input is in real format, so 2048 points are needed to compute one transform. Since the second fft is enabled, there is no reason to store old fft1 transforms. (This may change in the future.) Setting the first FFT amplitude to 30 is a good idea only if the radio hardware has much less dynamic range in the audio section than the sound card. It is easier than attenuating the signal from the IC-706 with resistors and serves the same purpose.

Then set these parameters on the second mode parameter screen:

- First backward FFT version [0]
- Sellim maxlevel [6000]
- First backward FFT *att. N* [4]
- Second FFT bandwidth factor in powers of [0]
- Second FFT window (power of sin) [2]
- Second forward FFT version [0]
- Second forward FFT *att. N* [7]
- Second FFT storage time [15]

If your computer is modern enough to support MMX instructions, there is no need to use them when processing a single channel that is sampled at 8-kHz only; so it is safe to set fft versions to 0. If your computer is better than a Pentium I, you may as well set the highest number. The *att. N* parameters control how the bits are shifted to prevent overflow or quantization noise. This is one of the complications coming from use of 16-bit arithmetic to save CPU time. It is completely useless in this case, but the code is optimized for other more demanding tasks. This link gives information about the *att. N* parameters and how they influence the processing: **antennspecialisten.se/~sm5bsz/linuxdsp/install/dlevel.htm**. Enable AFC/SPUR/DECODE on the next screen, and select the default parameters on the screen that follows.

Then set the final mode parameter screen like this:

- First mixer bandwidth reduction in powers of 2 [1]
- First mixer no of channels [1]
- Baseband storage time (s) [30]
- Output delay margin (0.1 s) [3]
- Output sampling speed (Hz) [8000]
- Default output mode [1]
- Audio expander exponent [3]

The baseband will be represented as *I* and *Q* with a sampling rate of 2 kHz. Make the bandwidth reduction larger if you want to save memory or CPU time when playing with really narrow filters. Notice that 1 kHz is the largest bandwidth you can have in the

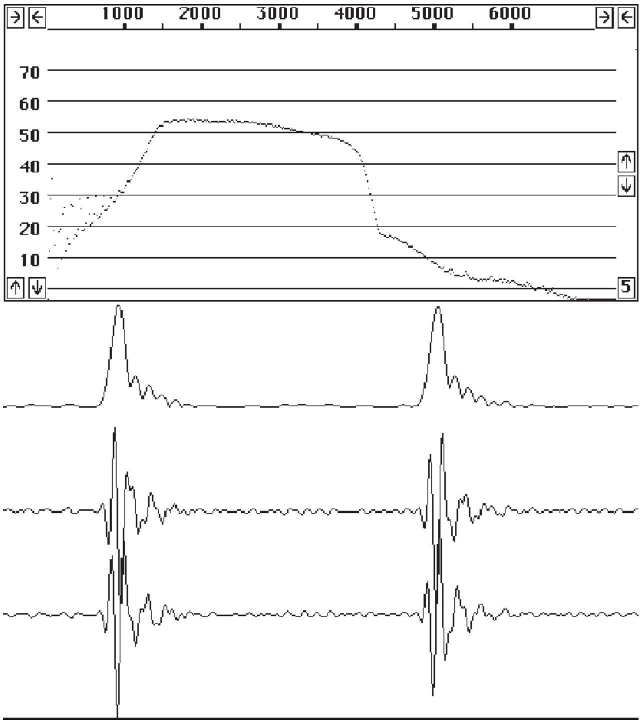


Fig 5—Frequency response and pulse response of the IC-706 as measured with *Linrad* in the uncalibrated state.

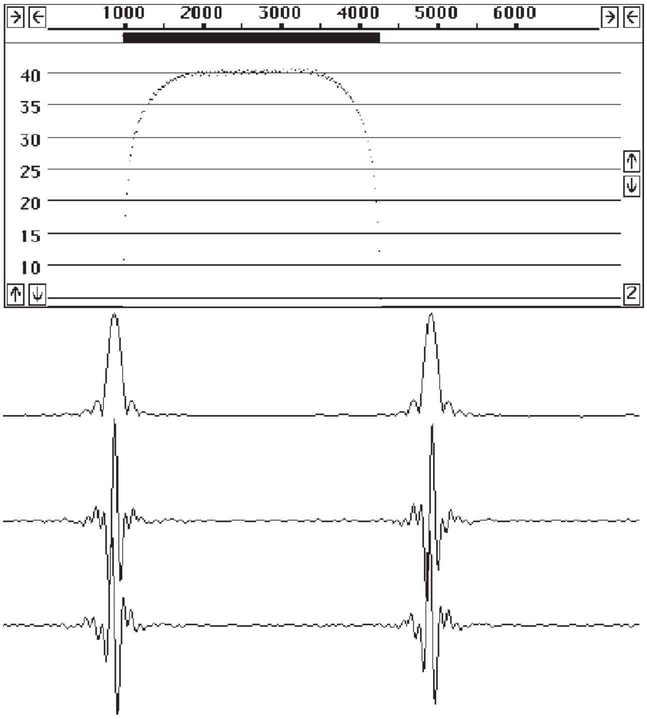


Fig 6—With one more filter in the signal path *Linrad* converts the frequency response to the curve selected by the operator during the calibration process. The pulse response is made symmetric and as short as possible with the skirt steepness selected.

output with the above settings, which maximize the output bandwidth. *Linrad* never gives more output bandwidth than 25% of the input bandwidth. If you want to process SSB signals, you must oversample the input at, for example, 24 kHz.

When set up as suggested here, *Linrad* should run with a CPU usage of about 22% on a 200-MHz Pentium MMX and 75% on a Pentium at 60 MHz. With the above parameters, 16 MB of memory is sufficient.

The *Linrad* Calibration Procedure

Besides the various spectra that display the signal in the frequency domain, *Linrad* has oscilloscope functions that display signals in the time domain. Fig 4 shows the result of feeding pulses into the IC-706 in both the frequency domain and in the time domain. In the frequency domain, we just see the frequency response of the signal reaching the loudspeaker. The filter bandwidth is 2.7 kHz at the –6 dB points if the slope of nearly 10 dB across the passband is accounted for. The lower amplitude towards the upper edge of the passband does not affect receiver performance, and it does not matter whether it originates in the IF filter or comes from the audio section. It is a matter of taste how the operator wants bass or treble set in the audio section and presumably ICOM has adjusted it to fit the built-in loudspeakers in a way that is generally acceptable.

There is only one problem with a sloping frequency response like that of the IC-706. If one measures the minimum discernible signal, MDS, in the way adopted by ARRL Lab, one must measure the frequency response and evaluate the correct noise bandwidth, which is a bit less than the filter bandwidth. The noise floor (in dBm/Hz) is constant across the passband regardless of the audio response.

The pulse response in the time domain is the Fourier transform of the frequency response. There is a lot to say about Fourier transforms in general; it can be found in mathematical textbooks. The transform of a soft function will be sharp and vice versa, for example. The steep edges of the filter at about 1 kHz and at about 4 kHz causes oscillations at 1 and 4 kHz in the time domain, while the wide flat region in the frequency domain corresponds to a short pulse in the time domain.

The oscilloscope traces of Fig 5 show the time-domain signal after it has been converted to a complex signal pair (*I* and *Q*) at half the original sampling speed. In the time function, the two lower tracks are *I* and *Q*, respectively, and the upper track amplitude is $(I^2 + Q^2)^{1/2}$. Most of the pulse energy is an oscillation at 2.5 kHz that lasts about one-and-a-half cycles. This is the energy from the essentially flat region of the passband. After the main structure, there are oscillations at about 1 kHz and about 4 kHz. These oscillations decay about five times more slowly than the main oscillation because the skirt steepness corresponds to a filter with five times higher *Q* than that associated with the filter bandwidth. The oscillation at 4 kHz is about 10 dB weaker than the oscillation at 1 kHz because of the 10 dB slope.

To show more clearly what happens, Fig 5 is produced with a sampling speed of 48 kHz, which results in a time function with eight times more resolution. This has no other good effects than making the oscilloscope traces easier to see. The CPU load increases and the calibration procedure becomes difficult because there is no pulse energy over most of the sampled frequency range.

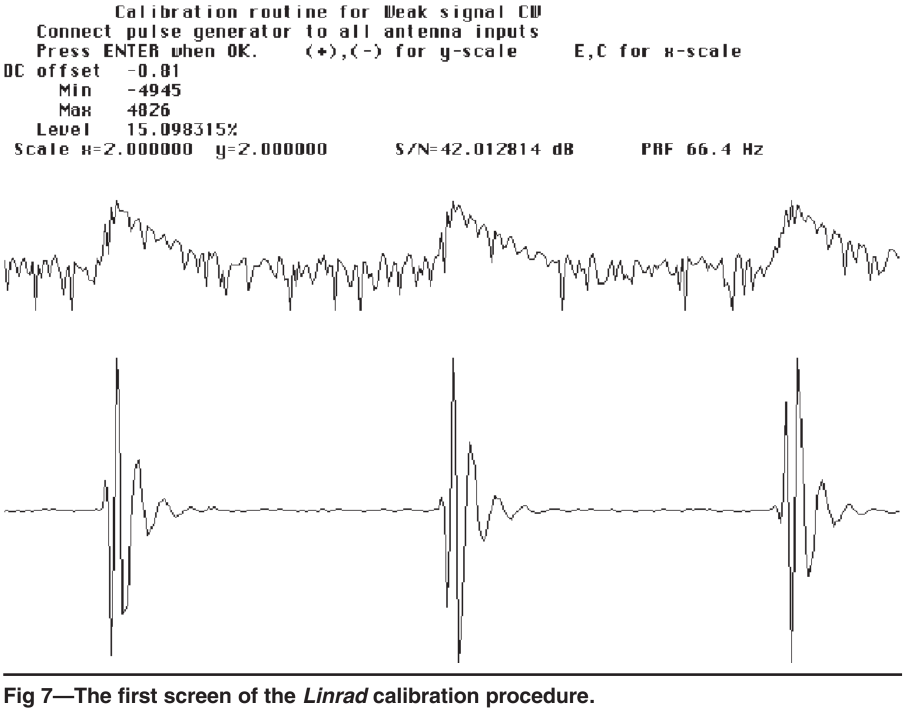
Fig 6 shows the time-domain and the frequency-domain responses after *Linrad* has been calibrated. The input signal is identical to the signal used for Fig 5. The pulse is still an oscillation at about 2.5 kHz that lasts for one and a half cycles, but the wider oscillations associated with the filter skirts are now symmetrical around the main peak, and the peak amplitude of the oscillations is about 6 dB lower. Notice that the frequency response is absolutely flat from 1.8 to 3.4 kHz. This is the frequency response associated with the time function shown in the oscilloscope tracings. Since the curvature of the filter is precisely known, it can be accounted for: The waterfall diagram is perfectly flat from 1.0 to 4.3 kHz.

It is up to the operator to select the frequency response. It is not very critical and something like Fig 6 is fine. The operator can choose because the *Linrad* noise blanker is still based on rather simple routines and 16-bit arithmetic is used to save CPU time. A very soft filter may cause quantization noise toward the spectrum ends, while very steep skirts may cause loss of accuracy due to overlapping pulses.

The first screen of the calibration procedure is shown in Fig 7. This screen is intended for adjustment of signal levels and pulse-repetition frequency. The upper track shows the power in a logarithmic scale, while the lower track shows the input signal in linear scale exactly as it is read from the sound card. The pulse-repetition frequency must be set low enough for the flat noise floor between the pulses in the logarithmic scale to be at least 50% of the total time. RF and audio volume controls as well as the amplitude of the pulse generator should be set to maximize the S/N of the pulses.

When the screen looks good, press [Enter] to start collecting an average of the pulse response. The next screen will show how the accumulated pulses look when translated to a frequency response.

As can be seen from Fig 7, the pulses cannot be averaged directly. They have a random phase and direct averaging will produce zero. Instead, the Fourier



```
           Calibration routine for Weak signal CW
   Connect pulse generator to all antenna inputs
   Press ENTER when OK.    (+),(-) for y-scale     E,C for x-scale
DC offset   -0.81
     Min    -4945
     Max    4826
   Level    15.098315%
 Scale x=2.000000  y=2.000000       S/N=42.012814 dB      PRF 66.4 Hz
```

Fig 7—The first screen of the *Linrad* calibration procedure.

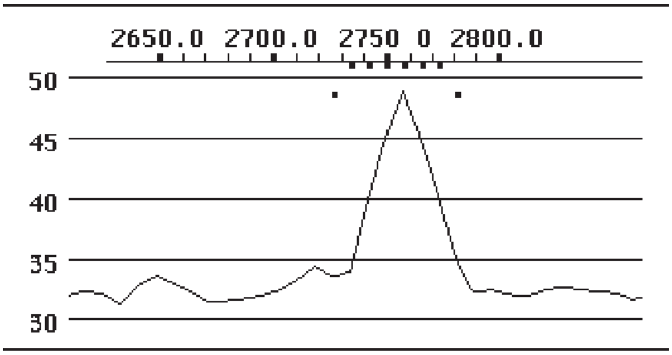


Fig 8—Baseband filter with 40-Hz bandwidth from a 256-point FFT. On screen, the size is indicated in the upper right corner as 8, the corresponding power of two, but that is not visible in this image. Carrier and sidebands of the CW signal are not resolved at this modest resolution.

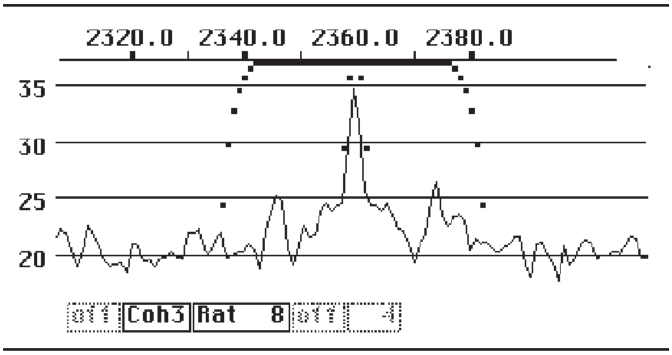


Fig 9—Baseband filter with 40 Hz bandwidth from a 2048 FFT. The signal filter is 40 bins wide, while the carrier filter is two bins wide. With about 1 Hz per fft bin, one can see the CW carrier and the principal keying sidebands.

transform is taken for each pulse. The amplitudes are averaged directly, but the phase is differentiated twice before averaging. The accumulated second derivative is then integrated twice to produce the average phase function. Read more about the details of the *Linrad* calibration procedure here at **antennspecialisten.se/~sm5bsz/linuxdsp/flat/flat.htm**.

Normal Operation of *Linrad*

When everything is set up, watch the waterfall graph. When something looks interesting, click on it and the corresponding signal is routed into the headphones immediately.

With the mode-dependent parameters described above, a typical bandwidth for CW could be 40 Hz. The baseband filter, like all other filters in *Linrad*, is implemented in the frequency domain, so it involves an FFT and an inverse FFT. The time delay through the filter is at least the time it takes to collect all the data points for one transform. With a 2-kHz sampling rate at baseband, the delay for a filter with 256 points is 0.13 seconds. This corresponds to a bin width of 7.8 Hz; so for a 40-Hz bandwidth, one needs six data bins over the flat region of the baseband filter.

Fig 8 shows the baseband graph with such a filter. When *Linrad* is set up like this, the processing delay is 1 s, which is about as much one can tolerate in normal CW traffic. Half of that is a margin that is not required on fast computers. It is set by the "Output delay margin" parameter. The little boxes in the upper-right corner set the size of the baseband FFT. One cannot set a number that is smaller than the size of the window, but the arrows in the upper-right corner can be used to expand the X-axis which will allow a smaller FFT size.

For extremely weak signals, one can switch to coherent processing. To do that, one needs a very narrow filter for the CW carrier so the baseband FFT size has to be larger. Fig 9 shows a typical filter for coherent CW. Here the baseband FFT size is set to 2048, which causes a delay of 1 s for a total processing delay of 2 s. In case where the AFC must run with a delay, the total processing delay will increase correspondingly. Use the F1 (help) key to find out about the different controls in the baseband graph and read more about it at **antennspecialisten.se/~sm5bsz/linuxdsp/run/basebgr.htm**. This link gives some more information about coherent CW: **antennspecialisten.se/~sm5bsz/linuxdsp/demo/coheme.htm**.

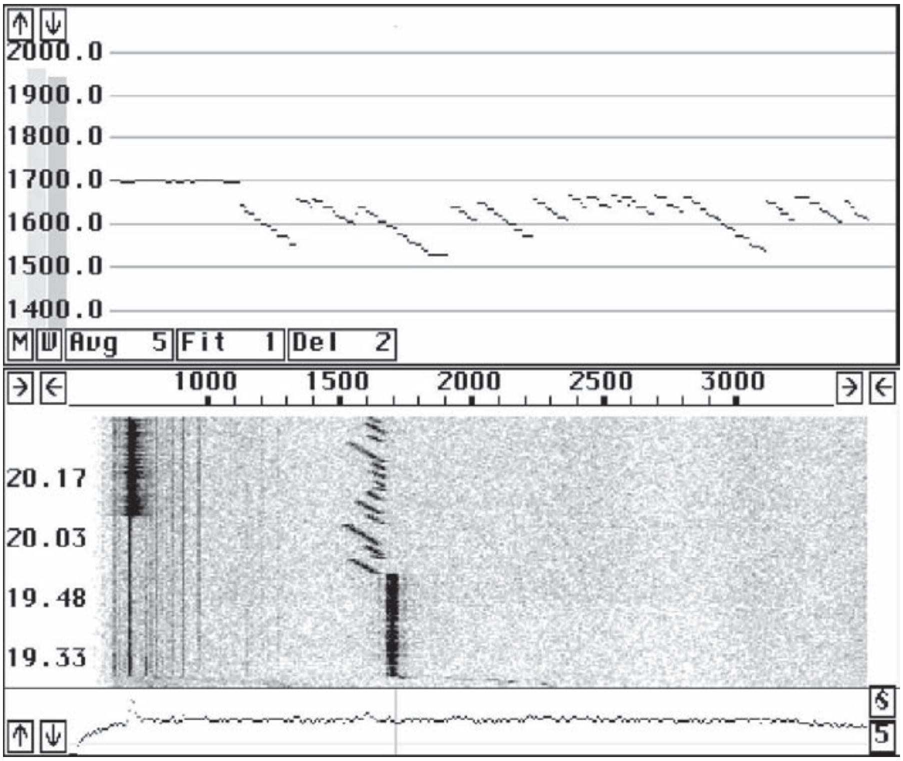


Fig 10—AFC graph (upper) and waterfall graph (lower) showing first a stable station, then a very unstable station drifting at about 1.5 kHz/min.

Using the AFC

It is well known that CW signals can be seen on a waterfall display at levels well below those at which they can be copied by ear. That means of course that the computer can locate CW signals precisely at signal levels below the detection threshold. This is used in *Linrad*. The AFC routine uses the same power spectrum used for the waterfall graph: The second FFT, if enabled, otherwise the first FFT is used.

The AFC is affected by the resolution at which the power spectra are available. The sensitivity increases with reduced bandwidth, but not proportionally, because wider bandwidths allow more averaging. Setting the FFT resolution narrower than the bandwidth of the signal does not improve sensitivity, but it does degrade time resolution for unstable signals.

The parameters suggested above give a second FFT size of 1024, which yields 4 Hz per bin and a bin bandwidth of about 8 Hz with a sine^2 window. This is adequate to keep signals centered in the desired passband within 1 Hz or so, and a signal that drifts by less than 100 Hz per minute can be located well enough by extrapolation of the frequency from several seconds back in time. That, in turn, means that the frequency is evaluated from typically 25 transforms, giving a S/N improvement of typically 7 dB.

For very unstable signals, the AFC can be run with very little averaging. Fig 10 shows the AFC graph and waterfall graph of a station that drifts by 100 Hz in 30 transforms. The mode parameters are set as described above, so that each transform spans 0.26 s. They overlap by 50% so they arrive at an interval of 0.13 s. The frequency drift is thus 25 Hz/s, which can also be read from the waterfall diagram, since it has both time and frequency scales. For the AFC, an average frequency is computed from five transforms or about 0.8 s. Over this time, the frequency drifts by about 20 Hz, twice the bin width of the FFT. In the waterfall, the averaging is four so it gives a good idea about the input data used to calculate the average frequency for the AFC.

Up to about 19.50, the stable and much stronger QSO partner is transmitting. The S/N is 25 dB in an 8-Hz bandwidth. On screen the AFC graph shows S/N in yellow, not visible in Fig 10. The stable station is at 1700 Hz. For the unstable station, S/N goes from about 17 dB to 13 dB during the transmission. The reason for the sawtooth frequency variation is that short interrupts are made in the transmission now and then. During these interrupts, the S/N falls to about 5 dB, which corresponds to the largest noise component with an averaging of only 2.5 times.

The AFC would normally fit a straight line to the average frequency so as to allow a more precise frequency determination. This will fail for a signal that makes abrupt frequency jumps, therefore the fit parameter is set to 1, which means that the average is used directly. The delay parameter is set to 2, which means that the average frequency is used to process the signal that belongs to the midpoint of the time span from which the frequency is determined. With a S/N of 13 dB in an 8-Hz bandwidth, this signal is not easy to copy at high speed even if the real S/N is a bit higher; the peak is smeared a little by frequency drift. With the AFC keeping the signal at the passband center within a few Hertz or so, the 60-Hz filter that fits the keying speed can be used for reception; at least 200 Hz would be required without AFC.

The example in Fig 10 is not intended to suggest that there is a great advantage in going from 200 Hz to 60 Hz. A trained operator already has a very sophisticated "signal processor" in his brain with a very efficient AFC. The example is intended to show principles only. If the keying speed had been four times slower to fit in a 15-Hz filter, everything would work exactly the same but at a four-times-slower time scale. Then, going from 50 Hz to 15 Hz would be a very significant advantage because the human brain cannot go much below 50 Hz in bandwidth. If the instability is much worse, so the signal moves around by tens of kilohertz, as could be the case on microwaves, the AFC can be set to produce a readable signal at a modest bandwidth if the hardware has bandwidth enough to accommodate the signal.

The UNKN422 challenge at **www.af9y.com** is a really weak 144-MHz EME signal with severe frequency drift. The AFC graph when running this file through *Linrad* is well suited to describe how the AFC can be set up for very weak signals. The ultimate goal is to make the AFC follow the frequency well enough to allow coherent processing of signals far below the level where copying is possible, and then perform averaging on the coherent data. For coherent data, S/N grows in proportion to the number of averages in contrast to non-coherent averaging that improves as the square root of *N* only.

Fig 11 shows how the *Linrad* AFC operates on the UNKN422 challenge. The size of the FFT used to produce Fig 11 was 4096 with a sine^2 window. This means that each transform spans a time of one second and that the bandwidth of each FFT bin is about 2 Hz. The transforms are interleaved by 50%, so two transforms are computed each second. Fig 12 shows a waterfall diagram from the transforms used to produce Fig 11.

The AFC averaging parameter was set to seven, which means that a spectrum was calculated from the average of seven transforms spanning a time of 3.5 seconds. A new average spectrum arrives every 0.5 second, and from each one a frequency and an associated S/N value is calculated. The crosses in Fig 11 show these frequency values and the boxes show the associated S/N values.

Fig 11 has been manually converted to black and white. On the *Linrad* screen the frequencies are green dots while the S/N values are yellow dots. The AFC fit parameter was set to 20, which means that a frequency is obtained by fitting a straight line to 20 of the crosses in Fig 11. This linear least-squares fitting is done with each frequency weighted by its associated S/N value, which should maximize the

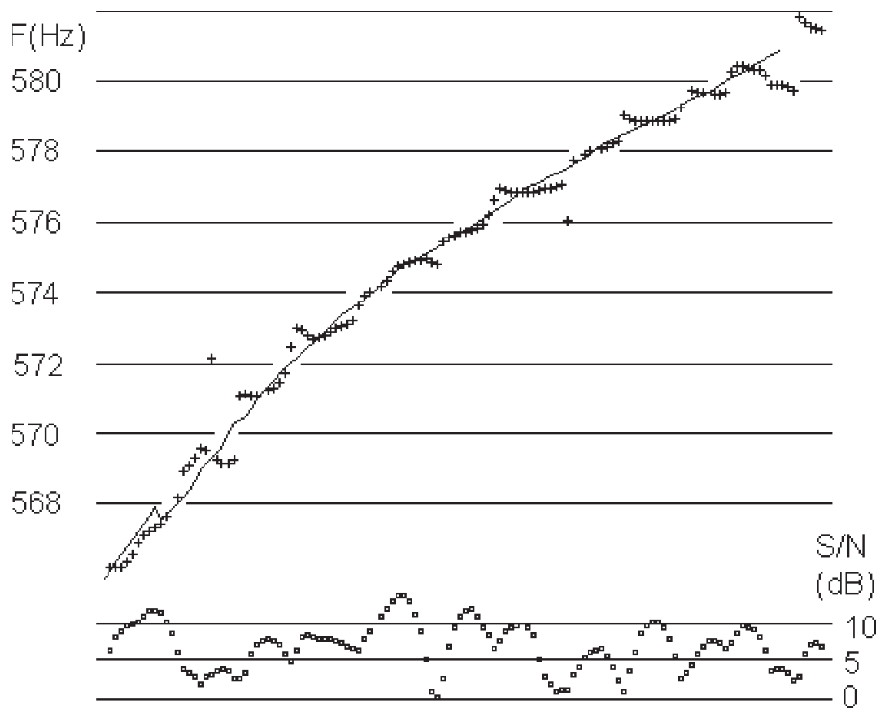


Fig 11—The *Linrad* ACF locked to the UNKN422.WAV file, a very weak EME signal.

probability that the fitted line is correct. This procedure means that the frequency-versus-time function is obtained over a 10 s period, and it does not matter that the signal is below the noise now and then, as can be seen from a comparison of Fig 11 and Fig 12.

The AFC delay parameter was set to 13, which means that the frequency of the straight line 6.5 s back in time is used to process the signal, which is 6.5 s old. This will give the best estimate of the frequency since each frequency value is based on what happened 6.5 s before and 6.5 s after the moment of time where the signal is extracted. The final AFC frequency forms the line in Fig 11 and a corresponding white line on the *Linrad* screen. The latest fitted straight line is shown in red on the *Linrad* screen, but that is not visible in Fig 11.

It is possible to set the delay parameter to zero for somewhat more stable or slightly stronger signals. The fitted straight line would then be an extrapolation and the AFC would not cause any time delay for the processed signal. An extrapolated frequency is of course less accurate, but the operator is free to choose the compromise between time delay and AFC accuracy that he or she finds best.

Spur Removal

Linrad has a procedure that locates very stable signals, spurs, which do not drift by more than a small fraction of the FFT bin width between transforms. This procedure is not yet automated; the user must point to each target signal. The frequency bin with maximum power is located and then a second-order polynomial is fitted to the phase-versus-time function from transform to transform. The phase information is then used to generate a constant-amplitude carrier in anti-phase, which is then added. The result is a very deep and extremely narrow notch filter. Since it is done in the frequency domain, it is very CPU-efficient. *Linrad* can process hundreds of spurs simultaneously and a CW signal that happens to come on top of a spur will be unaffected by the spur removal.

Baseband Processing

The desired signal is converted to frequency zero by means of a fixed frequency or by the frequency calculated by the AFC, if it is enabled. The sampling speed is reduced simultaneously as was described in a previous article *QEX* (May/Jun, 2003, pp 36-43). One of the interesting consequences of using AFC is that the spectrum of an unstable signal becomes narrower. This means that it is meaningful to make a new analysis of the power spectrum and use the improved S/N that can be obtained because of the narrower spectrum. When the AFC is enabled or coherent processing is selected, the baseband filter is not centered on the frequency zero, it is centered at the peak found in the averaged baseband power spectrum. Moving the filter to follow the carrier is a second AFC. It is controlled by the baseband spectrum averaging parameter *fft3 (avgnum)*, and it assures that the filter from which the carrier is extracted truly is centered on the carrier, even if it is only a fraction of a hertz wide. If the primary AFC is used, the time constant of it should be made longer than the time constant of the secondary AFC. The baseband window of the *Linrad* screen shows the frequency selected by the secondary AFC with a green cursor that will automatically place itself on the peak of the carrier. The yellow curves showing the filter in use are not moved around to reflect the filter position, that would be a waste of processing power in the event the process is run at higher data rates for high speed meteor scatter or some digital mode.

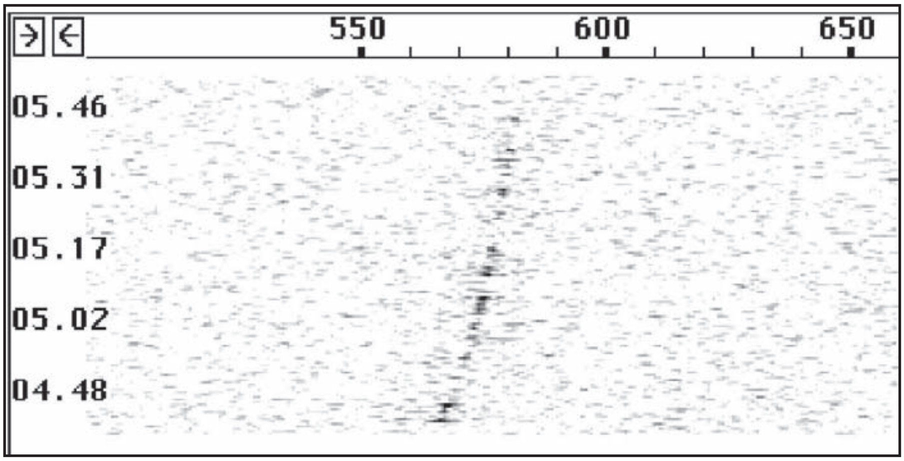


Fig 12—The transforms used to produce Fig 11 presented as a waterfall diagram. There is no averaging, each line is the power spectrum of a single transform.

The filtered baseband signal can be sent directly to the loudspeaker output after the frequency has been shifted by the amount specified by the BFO setting. It is possible to send the baseband signal to one ear and the carrier to the other. When doing this, one can make the carrier filter wide enough to allow the keying to go through but one can also make it narrower so it will be more like a phase reference for the ear receiving the keying information. Another possibility is to split the baseband signal into two components: *I* in phase with the carrier and *Q* 90° out of phase. One can elect to send these two signals *I* and *Q* to the two outputs to allow the operator to do the further mental processing. One can also elect to send *I* to both ears and skip the *Q* signal entirely.

At the present time, *Linrad* has only a mode for weak-signal CW. Therefore, there is no conventional AGC. The baseband signal is in a complex format, however. When it is limited to make sure it will not saturate the output, the phase angle is retained; the two components of the complex signal are attenuated by the same amount. This corresponds to RF clipping in a SSB transmitter and is equivalent to an AGC with immediate attack and release. A sine wave will not be distorted at all and the weak CW mode is quite useful for normal CW and for SSB despite the lack of a proper AGC function. In the future, I plan to add the conventional operating modes with AGC and noise-reduction algorithms that fit the larger bandwidth. When listening to weak signals, it may help to allow the signal to saturate in the conventional way, in the real-valued format of the output signal. The limiting of the baseband can therefore be disabled so one can get a sensitive measure of signal level by listening to the overtone content of the audio signal. Another possibility is to enable the audio expander, which will magnify amplitude variations of the baseband signal without introducing audio overtones.

Summary

This article has shown how *Linrad* can be used to improve performance of a standard Amateur Radio receiver both by improved noise blanking and by DSP filtering of the audio signal. The next article will focus on the high-end use of *Linrad* with two channels at large bandwidth. □□