Dither Explained

An explanation and proof of the benefit of dither

for the audio engineer

By Nika Aldrich

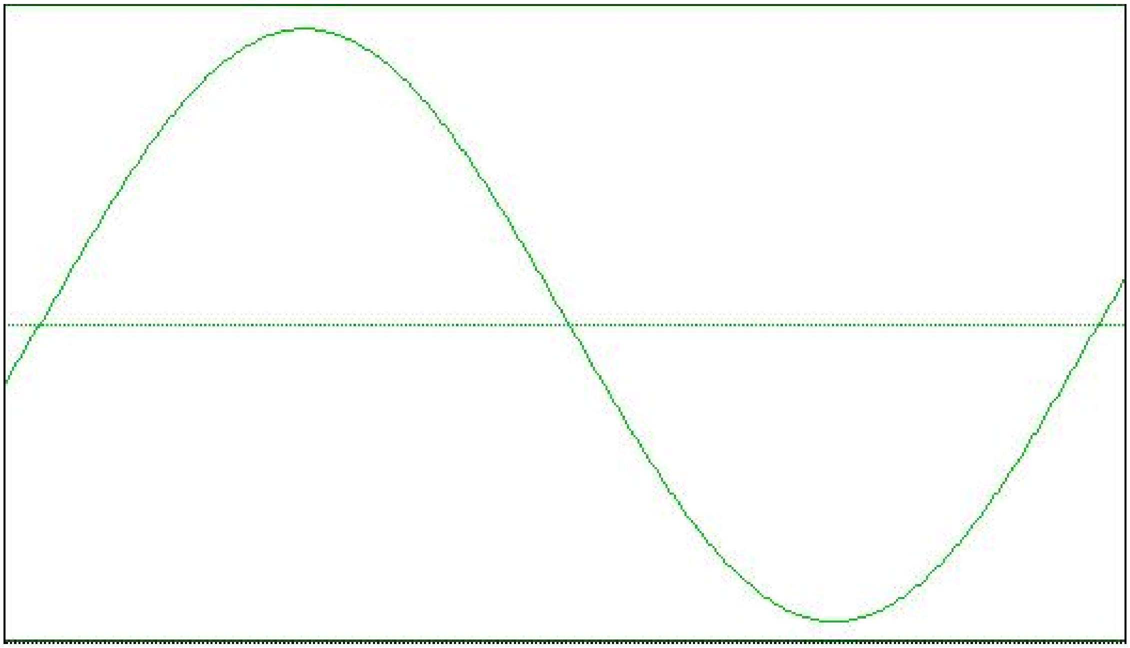
April 25, 2002

Several people have asked me to explain this, and I have to admit it was one of the most difficult concepts about digital audio for me to understand. It's not particularly intuitive: the concept of adding noise to reduce noise doesn't make a lot of sense.

It might help if we explain where the concept of dither comes from. In the 1940's the British naval air fleet were having problems with their navigation systems. The navigation systems were huge mechanisms full of cranks and gears and cogs. Everything was very primitive, kind of like an old grandfather's clock. Apparently the problem was that these cogs would chatter and stick and not move very smoothly, and would therefore be very difficult to calibrate. These problems seemed to go away when the airplanes were in the air. It was determined that the vibrations from the planes engines were in effect "lubricating" the cogs and gears and they were working more properly - predictably. This "noise" added to the system helped the accuracy of the system by removing the opportunity for the gears to stick. The result was that the British installed small motors on all of their navigation systems just to help vibrate the mechanisms on the ground (and perhaps more predictably in the air?) and everyone lived happily ever after. "Dither" is a British colloquialism for "undecidedness", or "wishy washiness", somewhat related to oscillation or vibration. The motors that they added to vibrate the navigation systems added "dither" to help a rigid mechanism operate more fluidly.

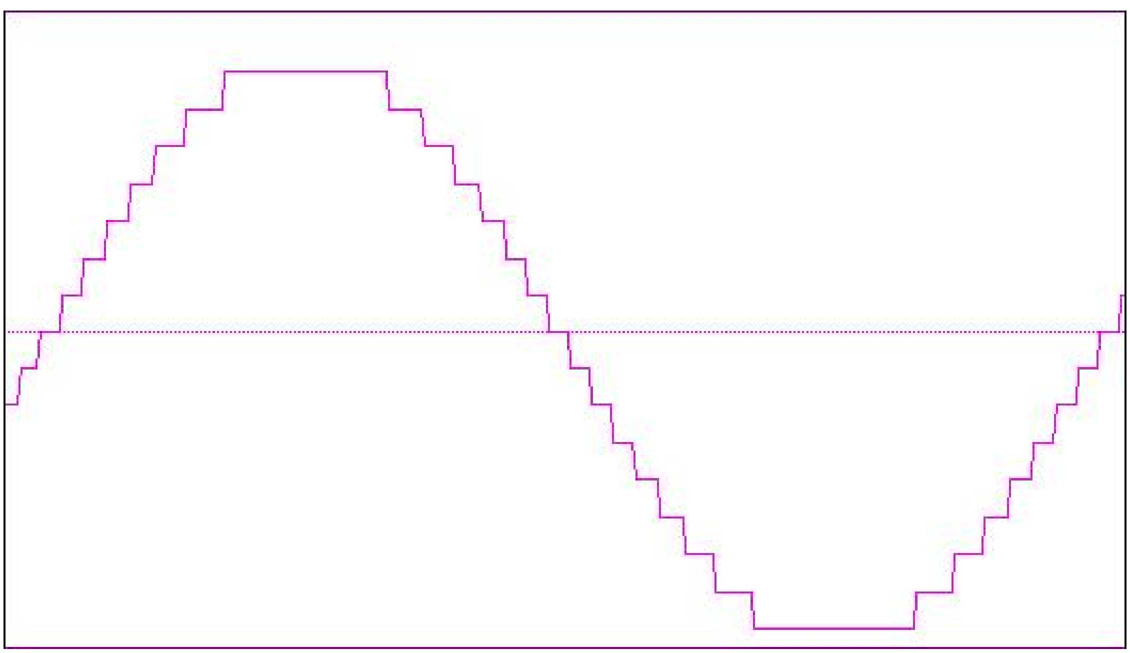
In audio what we're trying to avoid is the audio signal being "not fluid" because of truncation error. Whenever we go from a high resolution signal to a low resolution signal there is going to be quantization error as a result. In the picture below you'll see a 24 bit 100Hz sine wave recorded at low levels for the ease of visual demonstration and explanation.

Figure 1: 24 Bit 100Hz Sine Wave



The following is the same waveform that has been truncated to 16 bits. One will notice that because there are fewer quantization steps there is less opportunity for the signal to be "between" two points. The result is that there are stair steps where several consecutive samples end up being forced to be at the same level:

Figure 2: 24 Bit 100Hz Sine Wave Truncated to 16 Bits



You'll see that this waveform has been quantized such that there are a bunch of small squarish waves (it's actually a "stepped sine wave", but the properties and harmonic distortion are similar to that of square waves).

These stair steps in the waveform have an audible effect on the audio by adding determineable harmonic distortion to the signal at a fairly high level. Below you'll see two plots. The first plot is the original 24 bit waveform on a spectragraph (amplitude on the Y axis, frequency on the X axis). The second plot is the truncated version of this. You'll see in the truncated waveform a series of fairly high peaks at around 36dB down from the signal's peak. This is because this was a low level waveform to start with (the signal was recorded low to make the explanation more visibly obvious). If this were a "fully maximized" signal, (one that was recorded at full scale), these peaks would be much lower. Regardless, you'll see that in the 24 bit signal the noise is around 48dB lower. This is because the difference in dynamic range between 16bit and 24bit is 48dB. Obviously, 48dB improved dynamic range would yield a noisefloor 48dB lower on the same signal.

Figure 3: Spectragraph Plot of a 24 Bit 100Hz Sine Wave

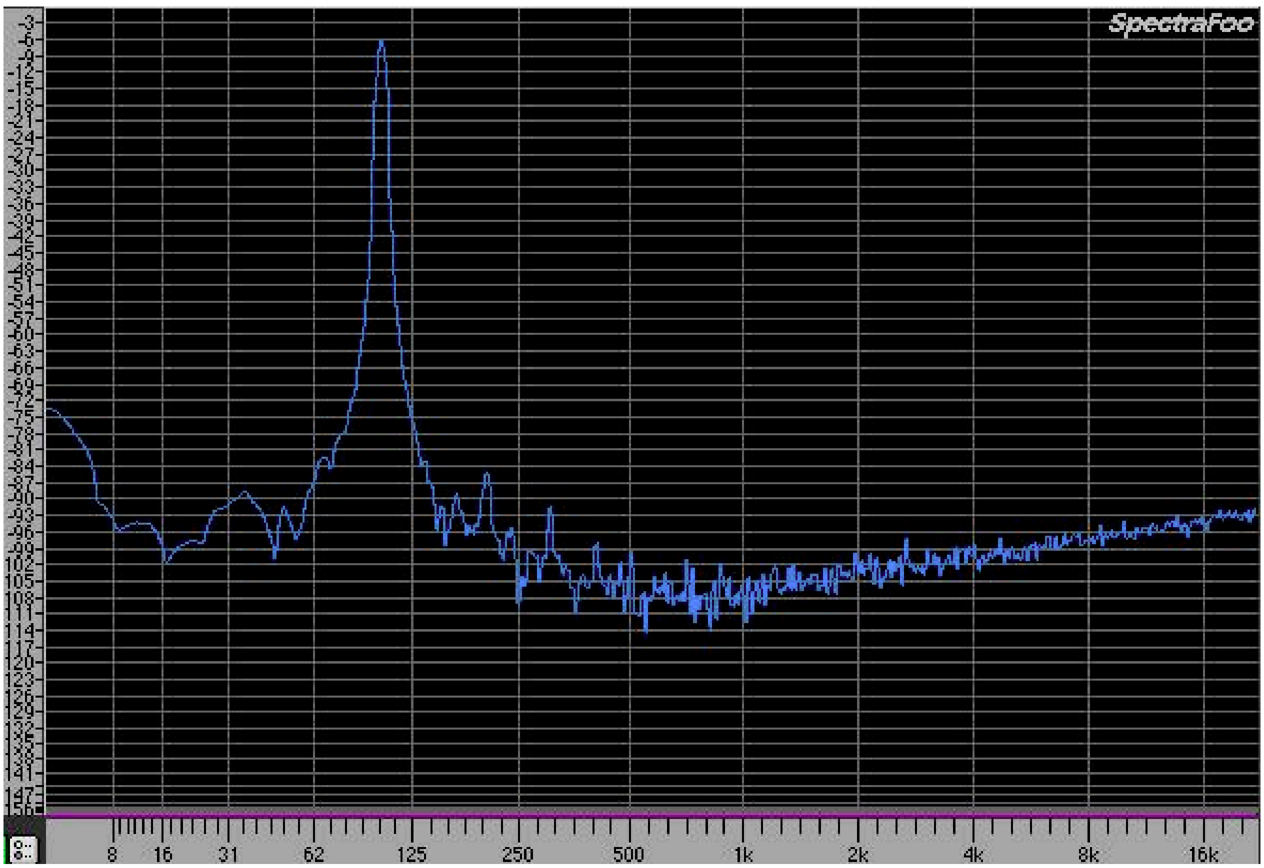
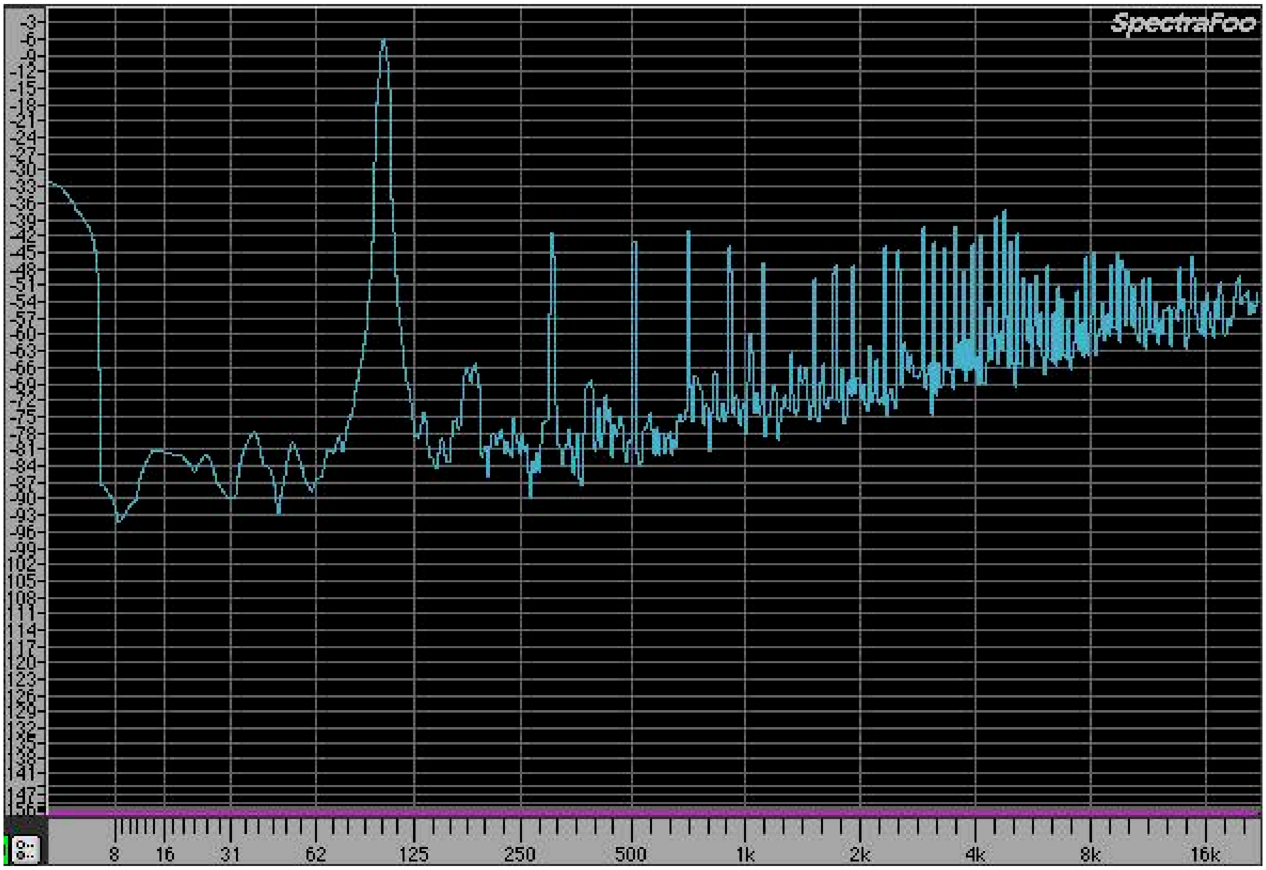


Figure 4: Spectragraph Plot of a 24 Bit 100Hz Sine Wave Truncated to 16 Bits



The concept of dither is to add some random noise to the waveform in order to "break up" the statistical determineability of the stair stepped waves. We do this by literally adding noise to the signal.

The noise is created by essentially randomly selecting bit values. In our working example we're dithering 24 bits to 16 bits, so we're going to add 9 bits of noise. This means that we're going to be randomly selecting numbers from 0 to 512 and adding those numbers to the original signal. We can look at it like we're going to add a random number from -256 to +256 to the signal. This series of numbers is randomly concocted by the computer. This set of random values will, in essence, create white noise. The random numbers, however, can't be totally random for reasons we'll get into, but first, in order to understand more about how we generate this noise we'll need to discuss "probability".

If you roll a dice you have an equal chance of getting numbers 1 through 6. This is called "rectangular probability". "Triangular probability" refers to a weighting in the probability of getting certain values. This is best demonstrated with two dice. If you roll two dice your odds of pulling up a certain number are as follows:

2: 1/1
3: 1/2, 2/1
4: 1/3, 2/2, 3/1
5: 1/4, 2/3, 3/2, 4/1
6: 1/5, 2/4, 3/3, 4/2, 5/1
7: 1/6, 2/5, 3/4, 4/3, 5/2, 6/1
8: 2/6, 3/5, 4/4, 5/4, 6/2

9: 3/6, 4/5, 5/4, 6/3
10:4/6, 5/5, 6/4
11:5/6, 6/5
12:6/6

This demonstrates triangular probability. You have the largest chance of rolling a seven, with triangularly decreasing probability of rolling other numbers so that you have the least odds of rolling a 2 or a 12. You can see why this is called "Triangular Probability" or TPDF ("Triangular Probability Desnity Function"). Natural white noise, however is what we call "Gaussian PDF" in that the distribution of the randomness follows that of a typical bell-shaped curve.

The reason we use Triangular PDF as the source of noise for dither is because we are stastically looking for a certain effect. Picture a DC signal at halfway between two quantization steps. We want there to be a 50% chance that it ends up on one quantization step or the other. If the DC signal is only 20% of the way from one quantization step to the other then we want a proportionate amount of quantization steps to end up at the one that the DC signal is closer to. Without going much further into this, the formula used to create the dither is most definitely mathematically determined to provide the most accurate results from a statistical perspective. Once the results are filtered at the D/A converter the end product will yield appropriate results.

Let's rewind and make sure we're caught up. White noise is created in a computer by randomly generating numbers that have the highest statistical probability of being "0" and the lowest statistical probability of being the minimum or maximum values (in our situation above that is -256 or +256.) Below is a picture of the waveform of the white noise that we're going to use for dither in this example. Below that is a spectragraph plot of that same noise. You can see that it indeed is white noise, or equal power at all frequencies (but on this logarithmic scale it shows greater power at higher octaves).

Figure 5: Dither Noise (RPDF)

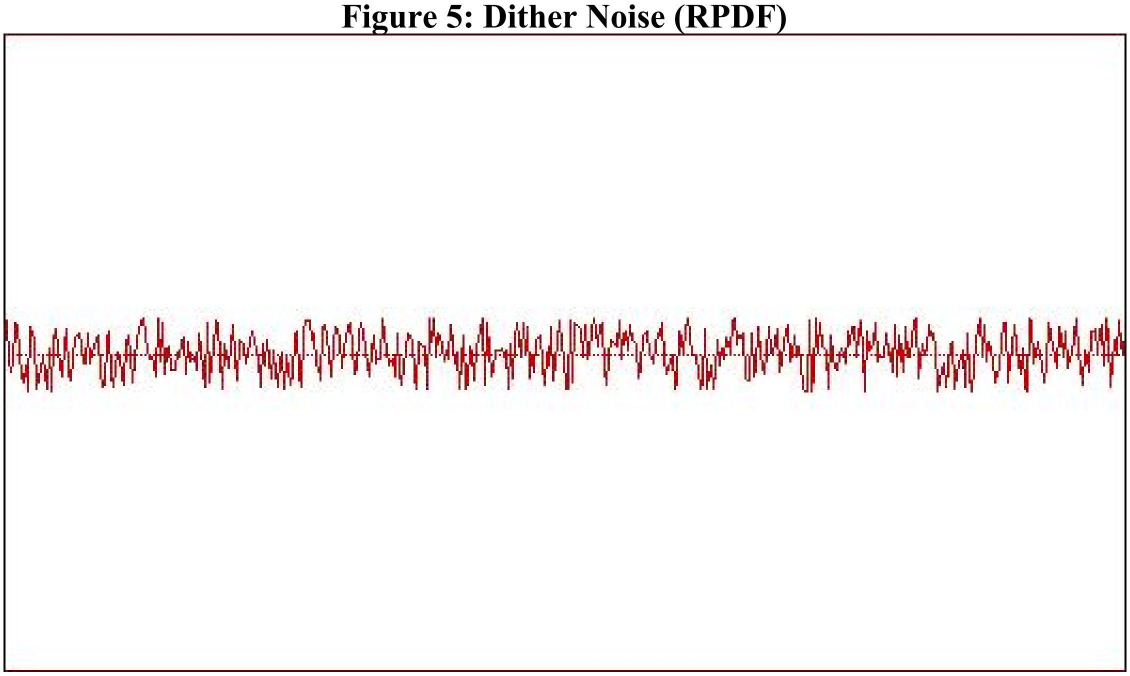
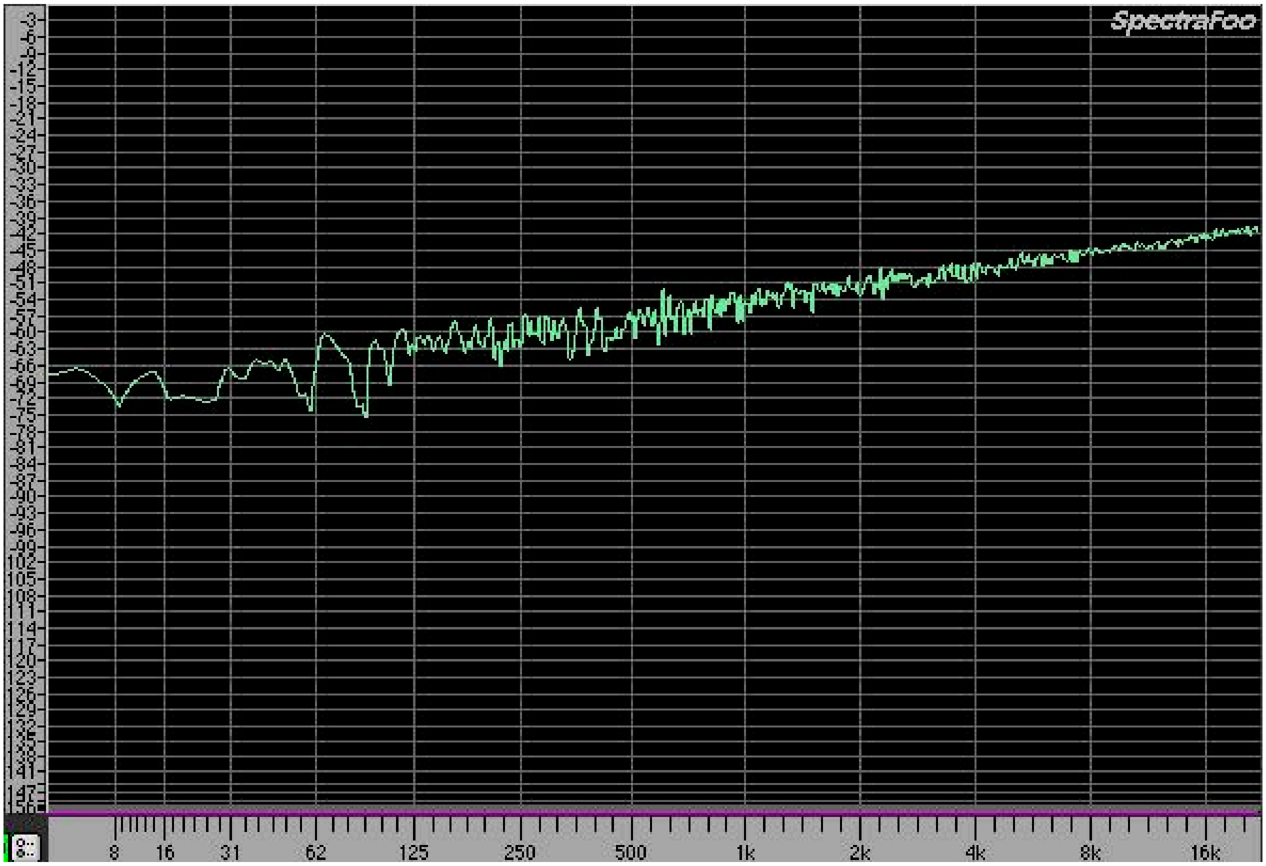


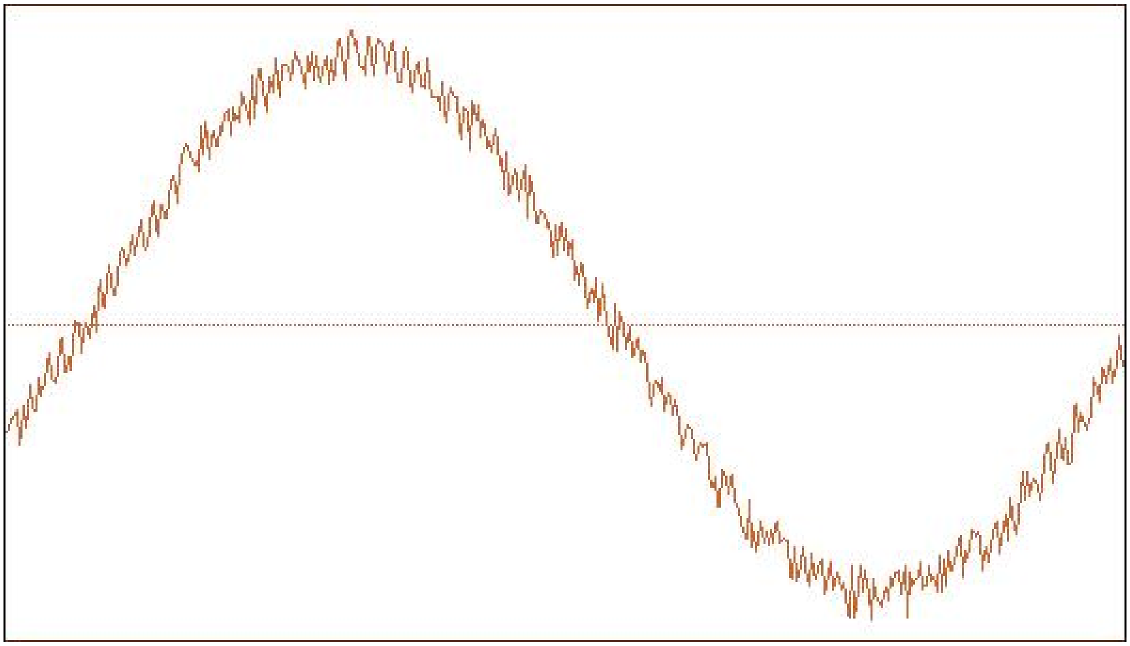
Figure 6: Spectragraph Plot of Dither Noise (RPDF)



One can notice that this is not true TPDF white noise, however, because with true TPDF white noise the majority of the plots on the waveform picture would be on or around the zero crossing. The white noise generator I happened to have handy for these pictures does not generate true TPDF, but more likely RPDF (rectangular probability), yet will still be adequate to show how dither works in this example.

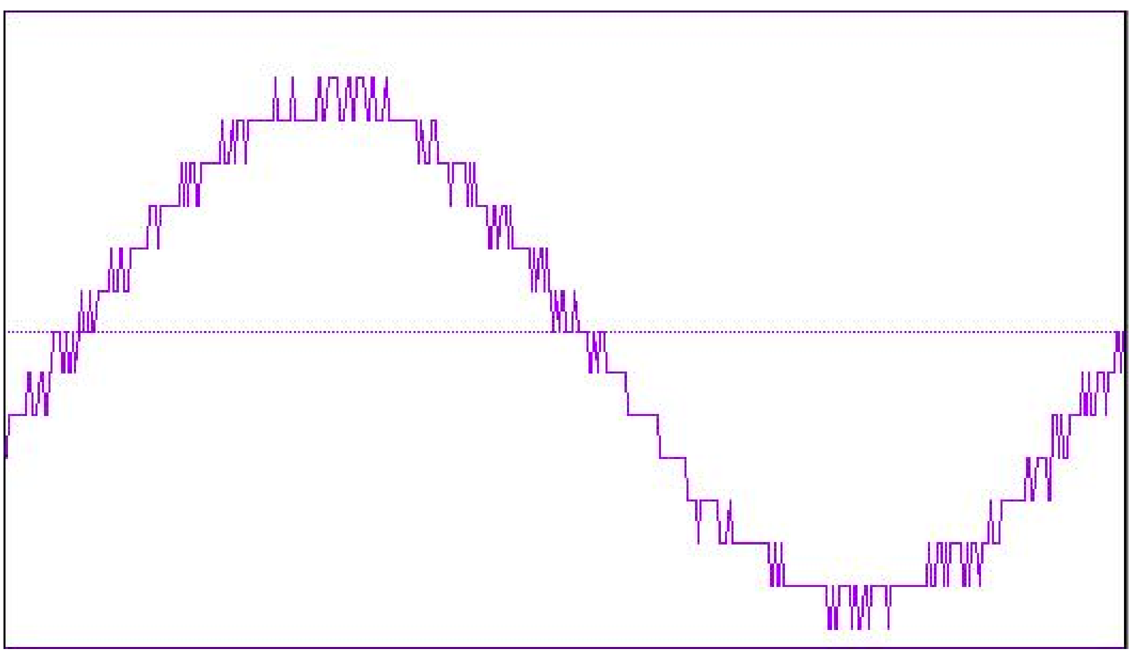
The following picture shows what happens when we add this dither to the original 24 bit untouched sine wave:

Figure 7: 24 bit Sine Wave with Dither Noise Added (RPDF)



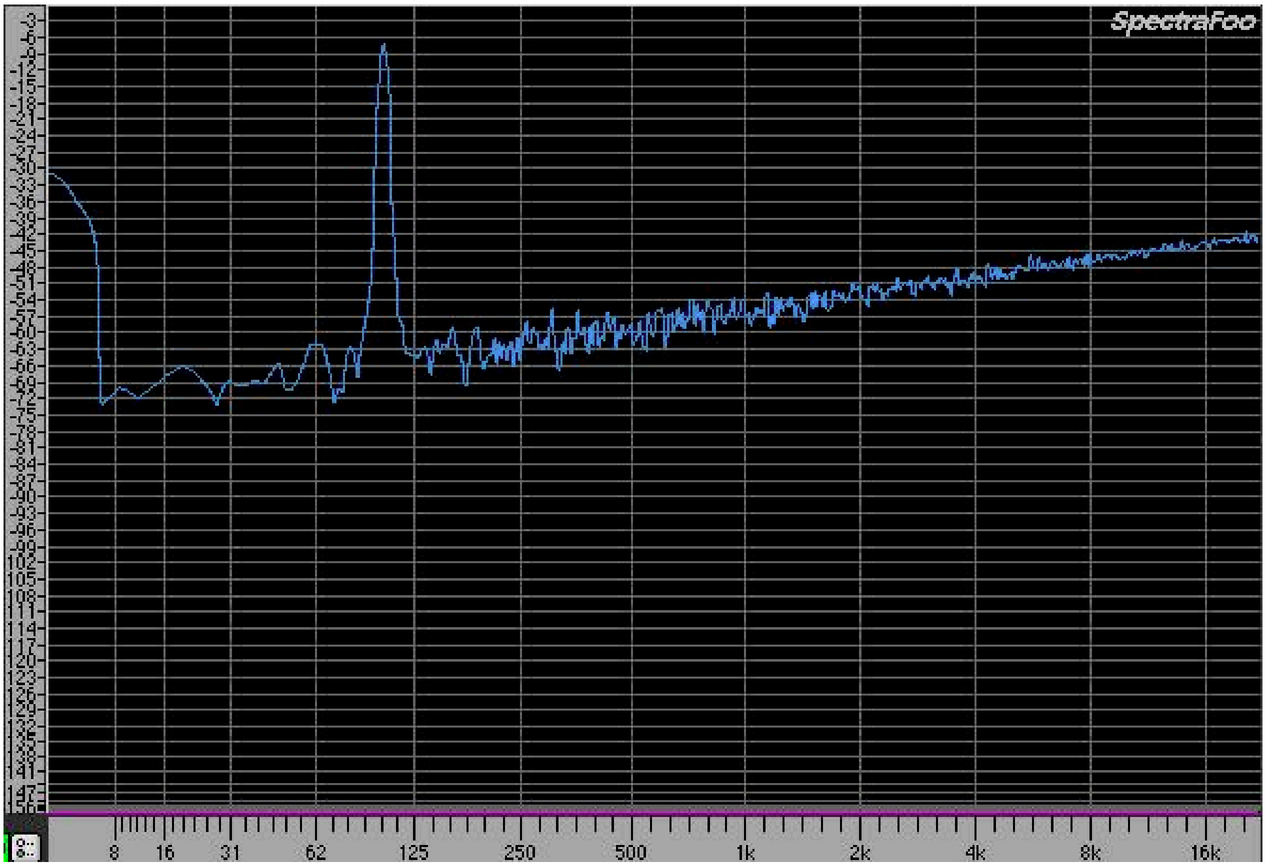
At this point we're still dealing with 24 bit words, but the last nine bits have had random noise added to them, and the results are evident in the picture above. When we now remove the last 8 bits to finally turn this signal into a 16 bit signal you'll see that we no longer have determineable "stair steps" as we did when just truncating, but we rather have random quantization fluctuations of single bits here and there.

Figure 8: 24 bit Sine Wave with Dither Noise Added (RPDF) Truncated to 16 bits



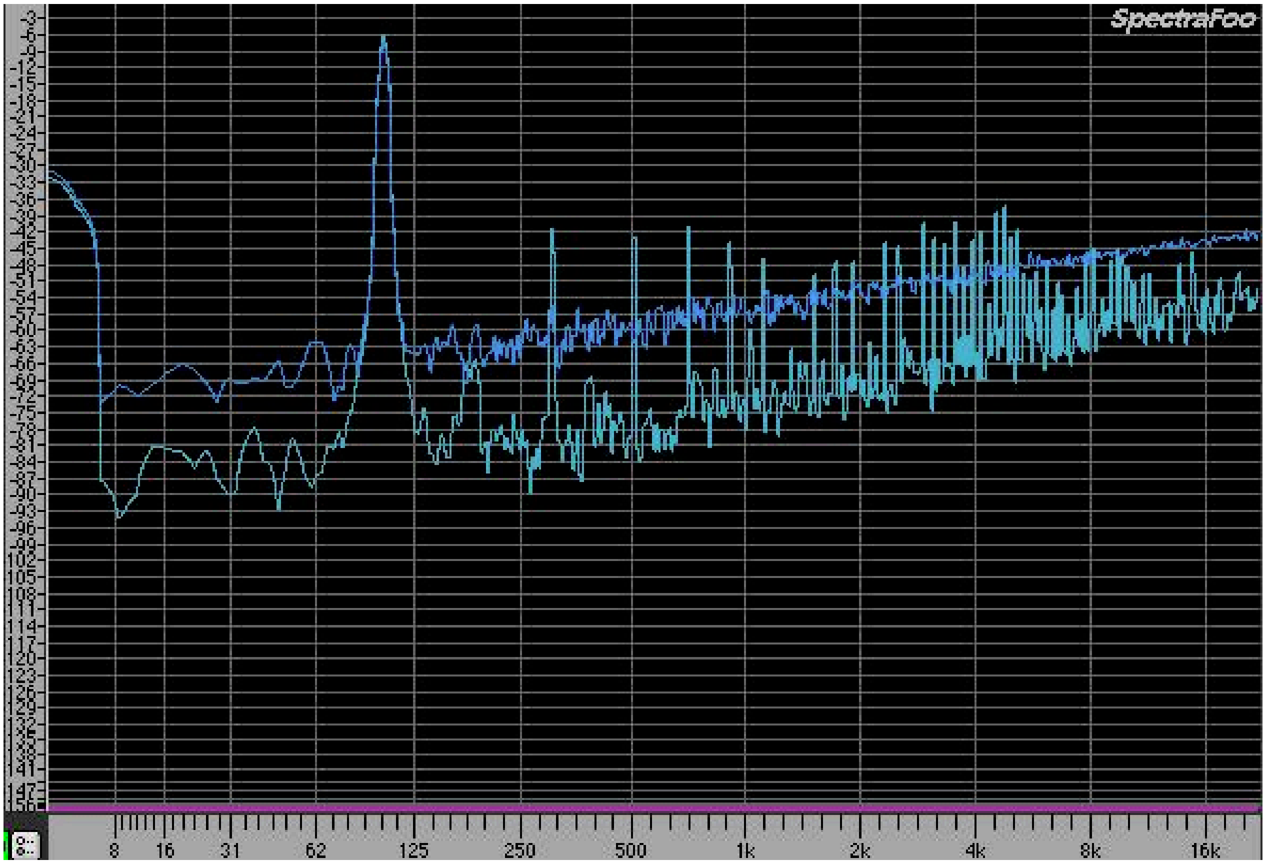
When we run this new signal through a spectragraph you'll see that we no longer have peaks in the noisefloor because of harmonic distortion. We now have a white noise floor instead. Yes, this white noise floor is higher than the original noisefloor from our 24 bit signal, but is a true white noise floor that is a natural occurrence, and therefore more pleasing to our ears.

Figure 9: Spectragraph Plot of 24 bit Sine Wave Dithered to 16 bits



When we compare this signal to our original truncated signal we'll see that the truncated signal has a lower noise floor for SOME frequencies, but the *peak* noise floor is much higher than the dithered version of the same signal. The dithered signal *effectually* has a lower noise floor by avoiding the distortion properties of those square waves. Below is a chart of the dithered and the truncated plots layered on top of each other.

Figure 10: Overlapping Spectragraph Plots of Dithered vs. Truncated Sine Waves

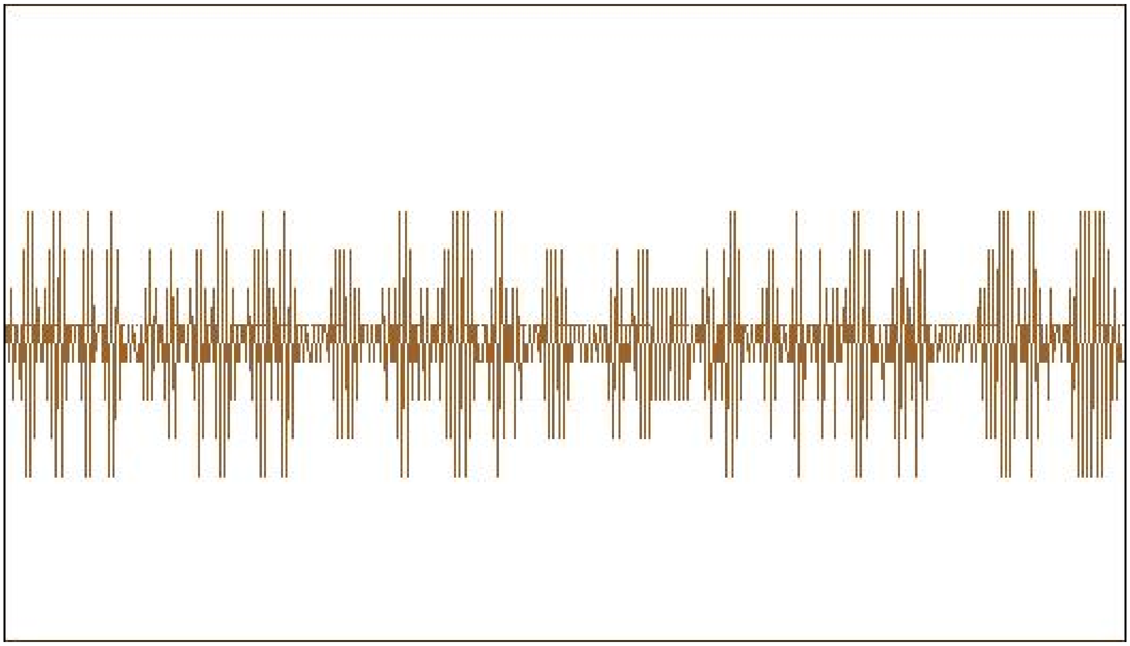


You'll see in the picture above that the sine wave peaks are at the same amplitude, but the truncated version obviously has a much less desireable result than the signal that we've added noise to to help randomize the signal and make it more natural sounding.

This is the crux of the discussion on dither. The bottom line is that WHENEVER a signal goes from a higher resolution to a lower resolution it is necessary to dither in order to avoid the artifacts provided by truncation that have been shown above. This means that whenever signals go from 48 bit resolution for processing to 24 bit resolution, or 24 bit resolution for mixing to 16 bit, or even analog (infinite resolution) to 24 bit during A/D conversion dithering needs to happen. In the case of the analog conversion process this dither happens naturally by means of the thermal noise within the converters Besides just the obvious effects of truncating there are also the cumulative effects from truncating a signal over and over again throughout processing. If a signal goes into an effects processor at 24 bits and gets inflated to 48 bits for processing but then leaves at 24 bits, and if this signal isn't dithered, then the truncation artifacts get added to the signal and could get multiplied if the signal goes through another similar stage of processing.

This all gets much more complicated, however, when we start to discuss other dither algorithms used for particular applications. The concept here is that it would be possible to add noise other than white noise that we humans might find even less offensive than true white noise. We humans are more sensitive to certain frequencies than others, so it would be possible to filter the noise so that it was more concentrated on frequencies that were less offensive. There are several algorithms on the market at this point from many manufacturers for adding unique dither to your audio. POW-r, DitherCD, and IDR are all examples of these types of algorithms. Apogee uses a scheme called "UV22" which focuses most of this random noise at frequencies above the human hearing range. The "22" comes from the fact that the majority of the noise occurs in their algorithm at 22kHz, right below the Nyquist frequency. Below is a plot of a waveform of UV22 noise.

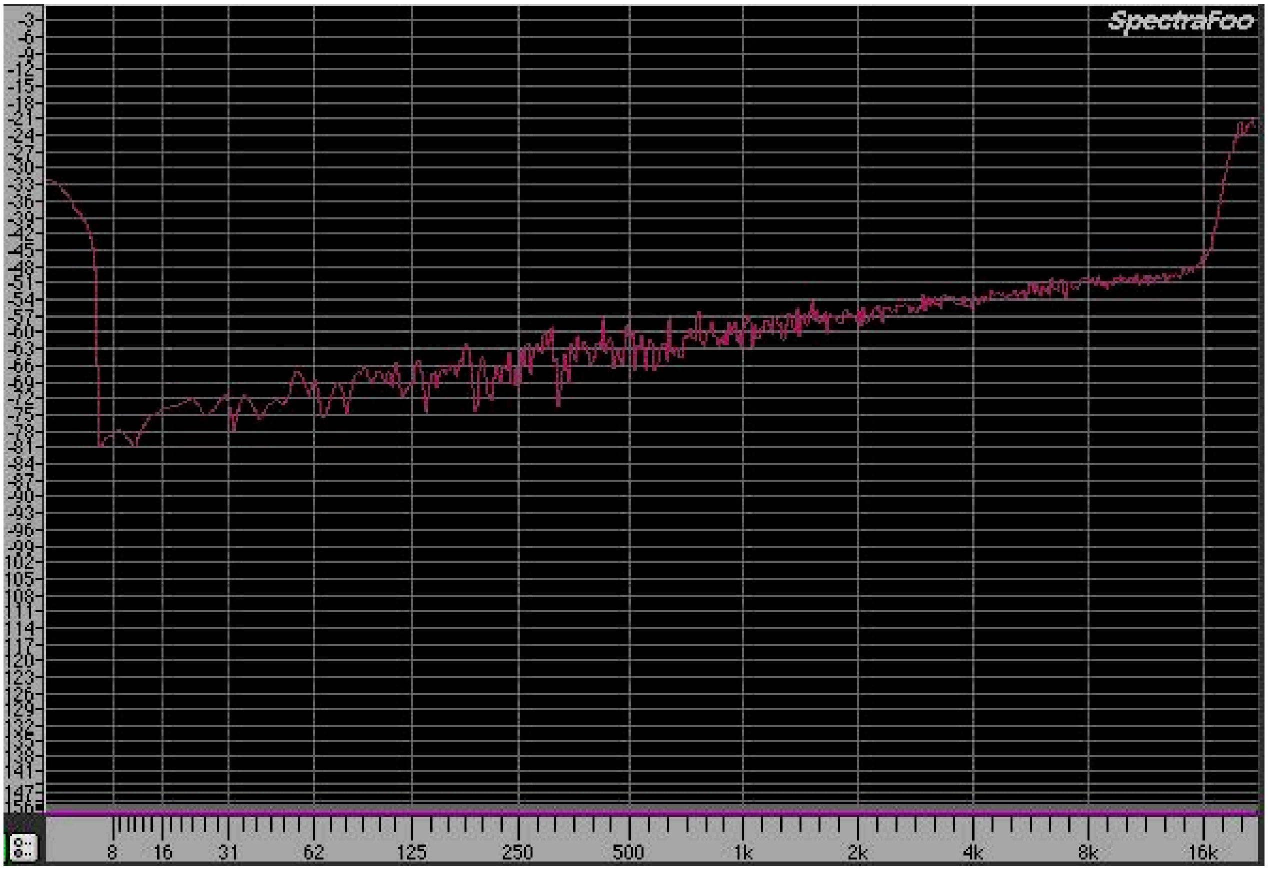
Figure 11: UV22 Noise



It is easy to see that, not only is this more closely resembling TPDF (as the majority of the points are at the zero crossing, but it is filtered toward higher frequencies as is indicated by the "narrowness" of the waveforms (which are presented in this chart and the similar chart provided above of dither noise in the same scale - 410 samples are shown). One other observation to be made is that the peak amplitude of this noise is much higher than the dither noise used above, but that amplitude is added only in the higher frequencies.

Below is a spectragraph plot of UV22 noise.

Figure 12: Spectragraph Plot of UV22 Noise



You'll notice a very steep rising pattern above 16kHz where the majority of the energy is focused.

The following is the same sine wave with UV22 added to it, and a plot of the same sine wave as above, noiseshaped (instead of dithered) with UV22. There will be two artifacts in the UV22 version (300Hz and 500Hz), but these are a result of errors in the way in which the graphs were created and are not a result of the UV22 process itself (for the curious, the sinewave had already been normalized, so the addition of the UV22 caused the waveform to clip, and the two spikes are indicative of this clipping).

Figure 13: 24 bit Sine Wave with UV22 Noise Added

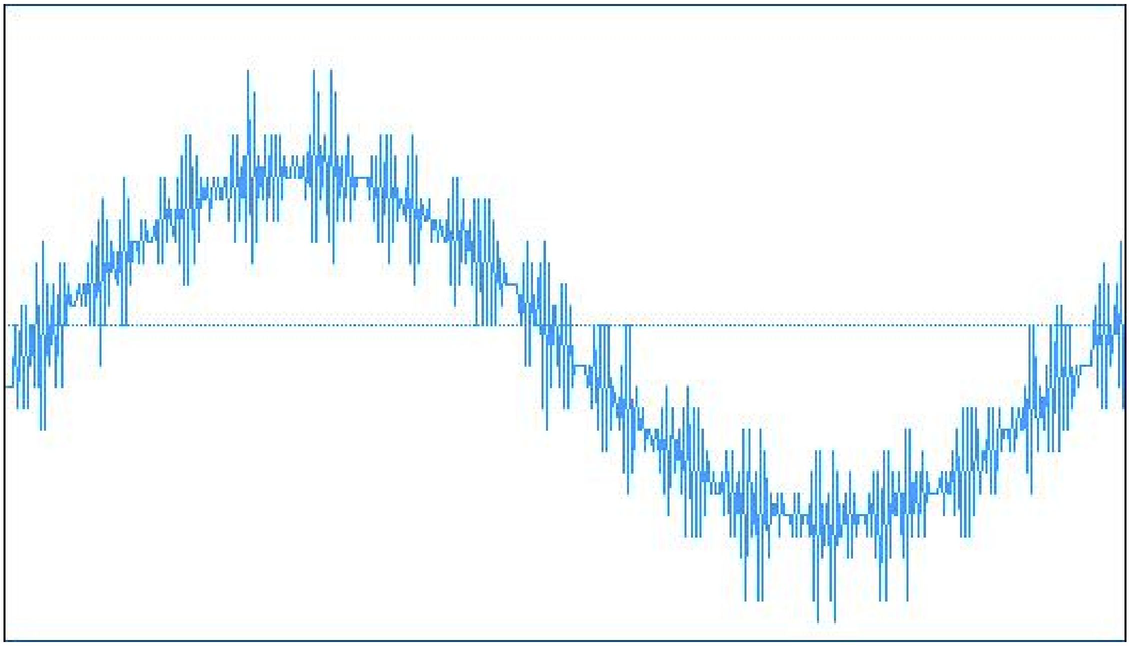
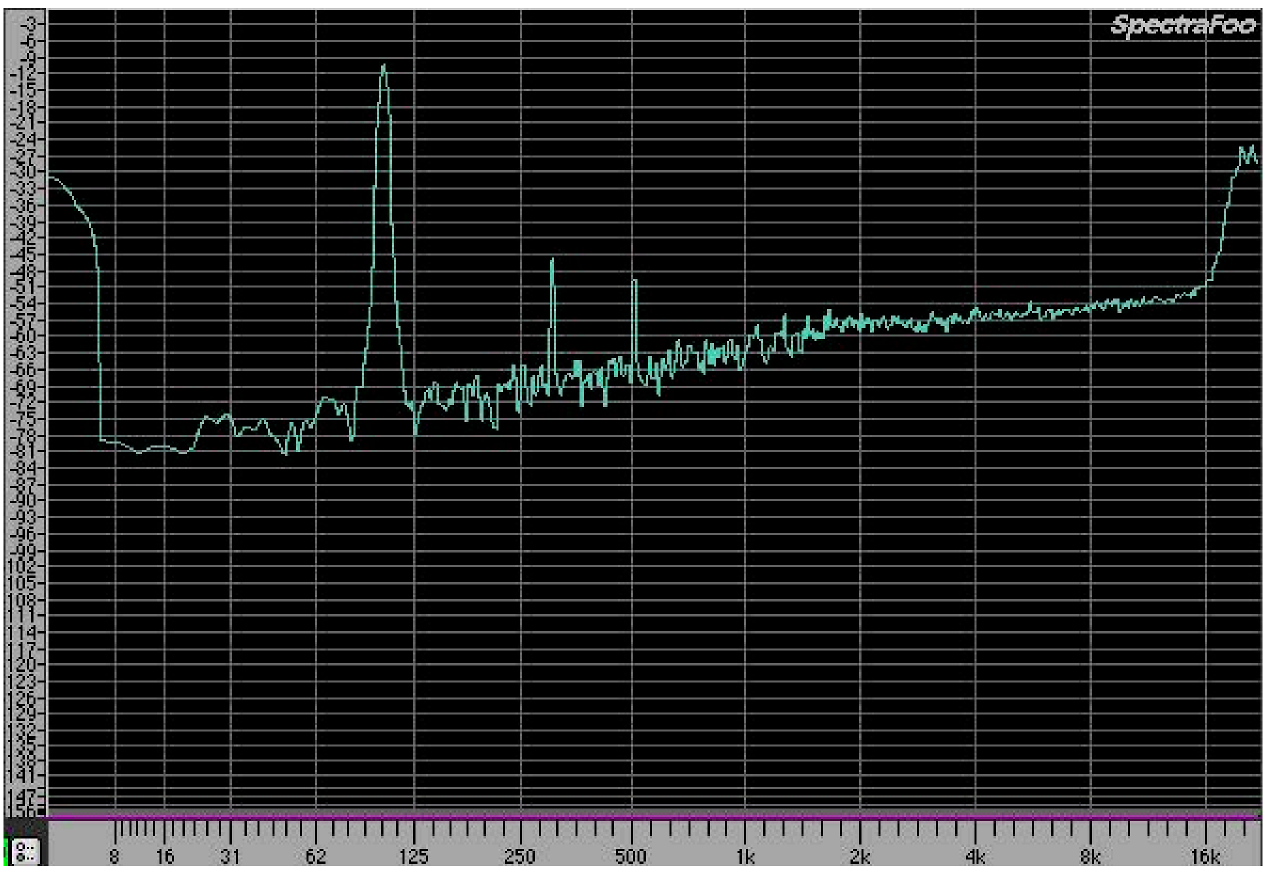
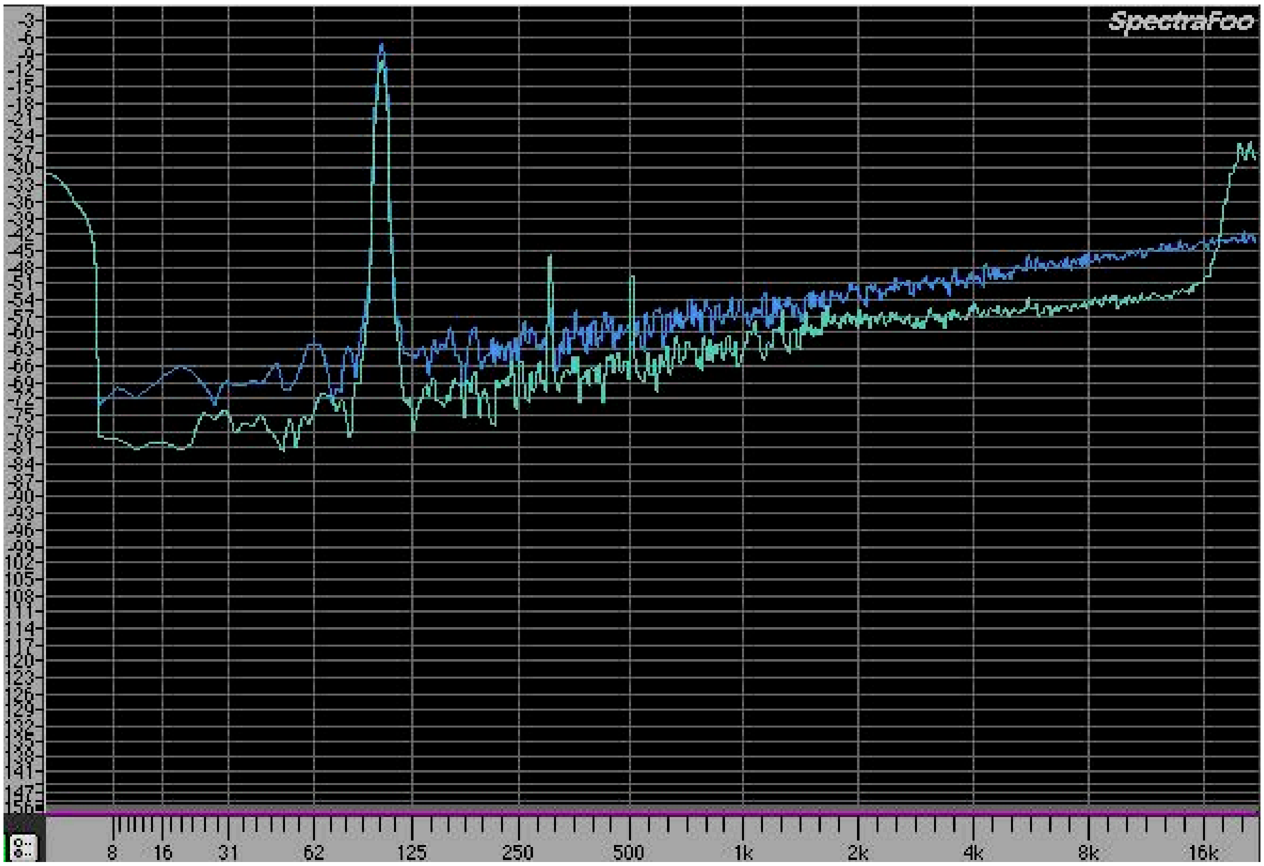


Figure 14: Spectragraph Plot of 24 bit Sine Wave with UV22 Noise Added



Below is a chart of the UV22'd waveform plot superimposed over the dithered waveform plot. Again, ignoring the artifically created artifacts you'll notice that the noisefloor is lower on the UV22 version within the audible range, while still having a higher peak noisefloor. The result is a lower perceived noisefloor and therefore greater dynamic range over the audible range for the signal.

Figure 15: Overlapping Spectragraph Plots of Dithered vs. UV22'd Sine Waves



This "colored dither", which is not true TPDF dither, is a process that should only ever be applied in the very final stage of processing. The noise that is used, while more audibly transparent, can end up "working it's way" into the audible range in unpleasant ways if this dithered signal is processed any further. As was mentioned above, there are several such specialized dither algorithms available on the market now, many of which have viable reasons for being beneficial. Listening (as is true in most areas of our industry) is the best way of determining your favorite dither algorithm, but it is important to remember that this is only for the FINAL processing stage. If your material is to be mastered then this should only be done at mastering. It should never be applied mid-mixing. TPDF is the only appropriate method that manufacturers should use for dithering after various processing stages, but it should ALWAYS be used after processing stages if the processing is at higher bit depths than the result is. Many software applications and hardware boxes provide options for dithering. The rules of thumb to follow are:

· If the signal has had any processing happen to it at higher bit depths then choose to add dither (there are some times that the option is available to add dither when no processing at higher bit depths was done, so this may not be as easy to do as it sounds).

· If ANY more processing is to happen to this audio then the appropriate dither to add is TPDF. If you are unsure which dithering option is the TPDF option then contact the manufacturer.

· If this is the final stage, immediately before pressing a master, other choices of "colored" dither may be used.

Dithering is a required part of the digital audio process if the results are to meet the expectations set forth by Nyquist and other digital theorists.

One last note is that the benefits of dither are more visibly apparent on lower frequency material than higher frequency material, however the benefits to higher frequency material can be noticed if observations over many cycles are made On higher frequency material the artifacts created by quantizing error can be above the Nyquist limit and fold back as aliasing into the audible bands.

I hope that this has been helpful. If there are any questions please ask. I'll probably end up posting this on my website and referring to it for people that have questions on this matter in the future. If I need to add anything, please let me know (I have already edited it more than once based on others contributing with additional information). If something is not clear please let me know that as well. I'd like to have it pretty cut and dried for future readers.

---

edited for accuracy 5.5.02