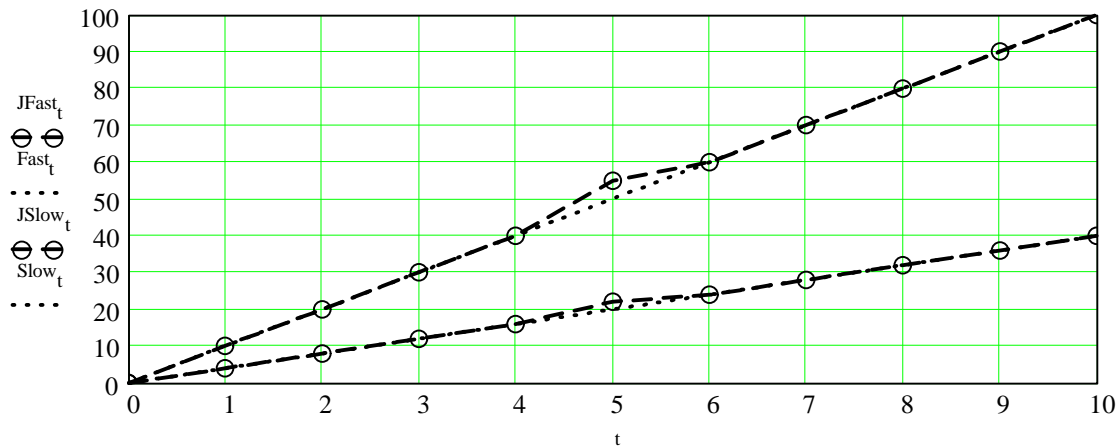


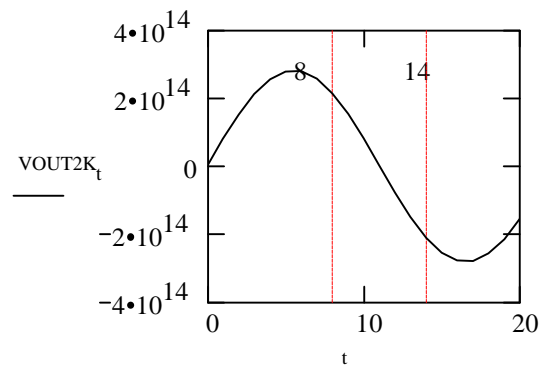
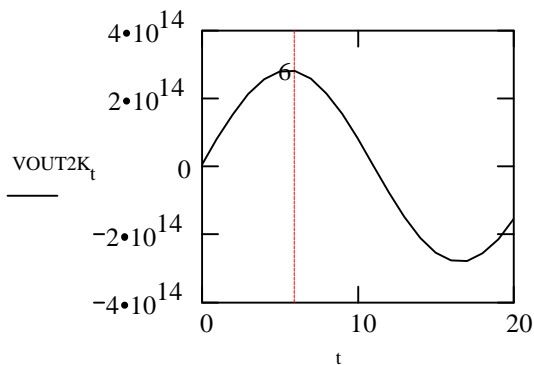
The first part of this article for people who want to gain some intuitive understanding about clock jitter problems, but wish to avoid the somewhat complicated mathematics associated with the subject. The second part deals with hardware related problems and practical considerations.

Understanding Jitter

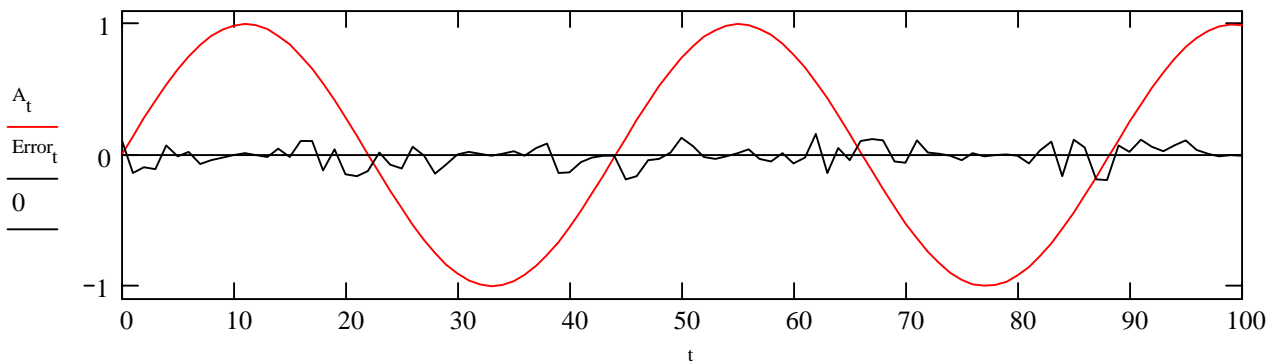
A time error in the sampling clock may cause a deviation from desired sample value. Such errors occur when the sample / hold circuit for an analog to digital converter is driven by an unsteady clock. Similarly, the reconstruction of an analog signal by a digital to analog circuit requires a jitter free clock. When sampling or reconstructing a DC signal, timing is unimportant, but fast changing signals may deviate too far during short timing errors. The plot below shows slow and fast sloping lines, with timing errors at sample #5. Note that the deviation from straight line is larger for the fast rising case.



When examining a sine wave tone, we can recognize the time portion of the signals that are less susceptible to jitter. The peak (left plot) as well as the dip are low slope regions. Conversely, there are time regions (right plot) of high susceptibility to timing errors. Maximum slopes occur during the highest frequency and the highest amplitude signals. Lower damage due to jitter occurs at lower frequency and amplitude signals.



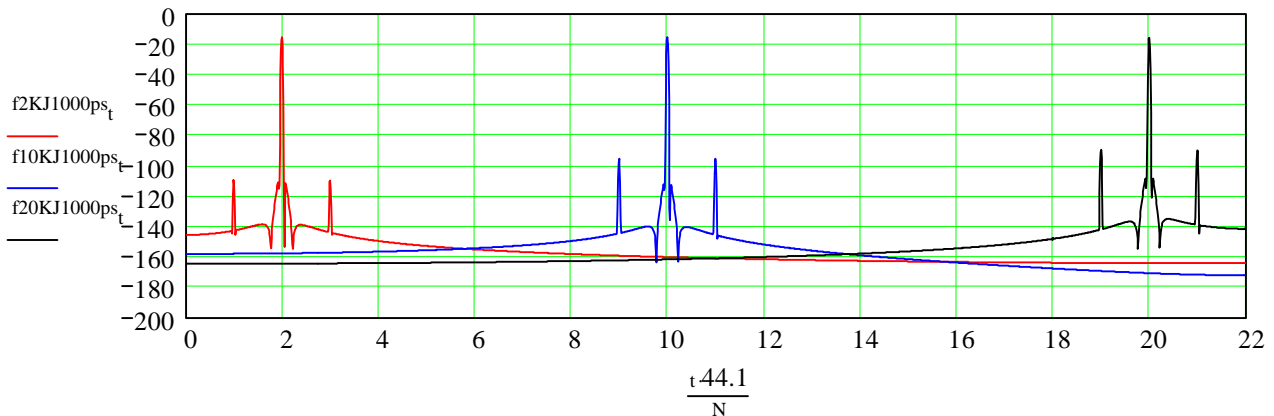
The next plot shows the outcome of introducing a random amount of time error to the sampling clock. Note that the error is small when the signal reaches peak (and deep) values. Larger errors occur at the high slope regions. The random sampling time errors interact with the signal, yielding a signal dependent error.



We now introduce a non random jitter error. The plot below shows the effect of a 1KHz sinusoidal jitter modulation on 2KHz, 10KHz and 20KHz tones. The plots are based on 1ns jitter amplitude (sampling frequency is set to 44.1KHz).

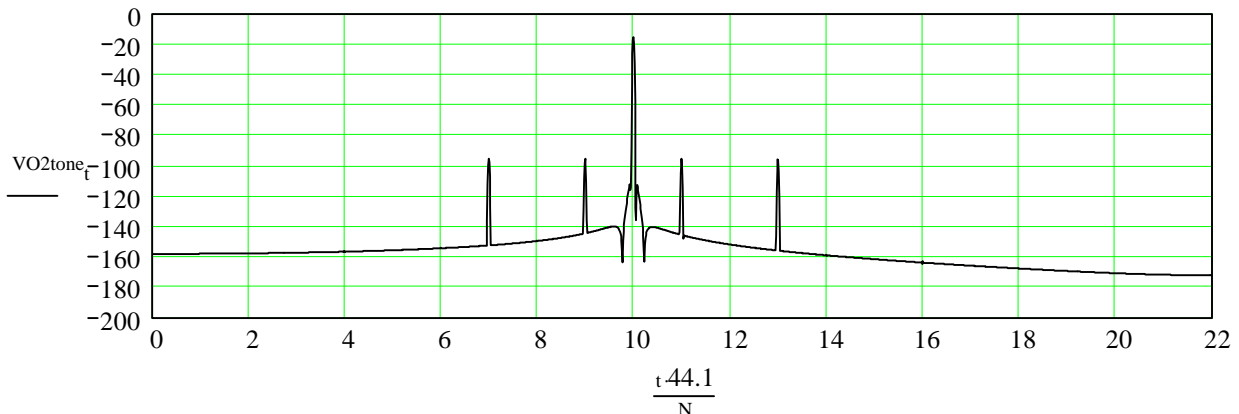
The jitter adds a pair of sideband tones to each of the signals. The sideband frequencies are 1KHz away from the tone (for the case of 1KHz jitter modulation). Note that sideband amplitude increases proportionally with increased tone frequency.

The horizontal axis is frequency and the vertical is a dB scale.



We next examine a plot showing jitter modulation by a dual tone signal, 1KHz and 3KHz.

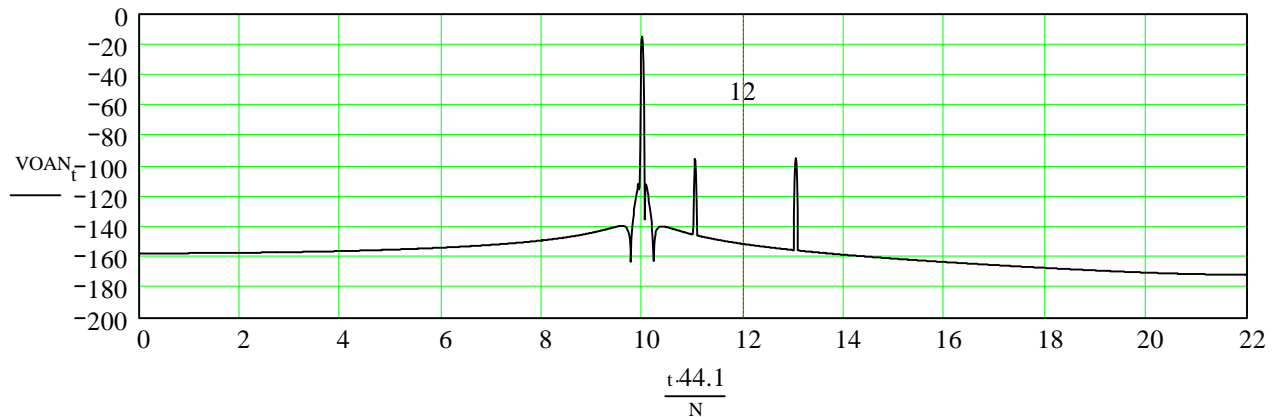
The sidebands frequencies are located 1KHz and 3KHz away from the 10KHz tone.



Increasing the jitter frequency increases the frequency interval between the tone and the sidebands.

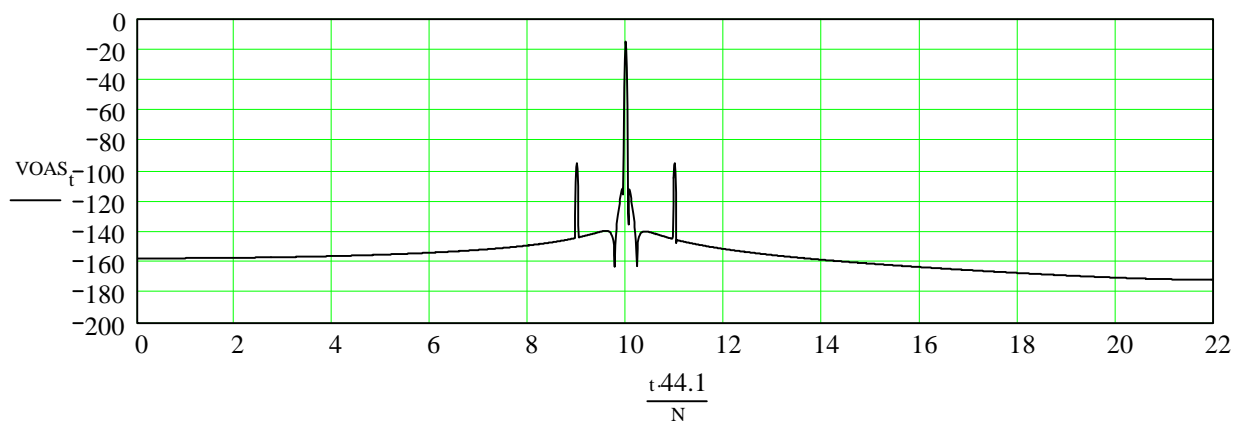
What will happen when the jitter modulation is above Nyquist? The next plot shows a 10KHz tone jitter modulated by a 23.05KHz tone (1KHz above Nyquist). The sidebands are located at 1KHz away from about 12KHz.

The "new center frequency" is $f(\text{Nyquist}) - f(\text{signal}) = 22.05 - 10 = 12.05\text{KHz}$



Let us raise the jitter tone frequency to 1KHz above the sampling frequency (45.1KHz jitter). The outcome is identical to that of the 1KHz case. One may view the "mechanism" as a time difference between two waves, one being the clock signal and the other an analog wave.

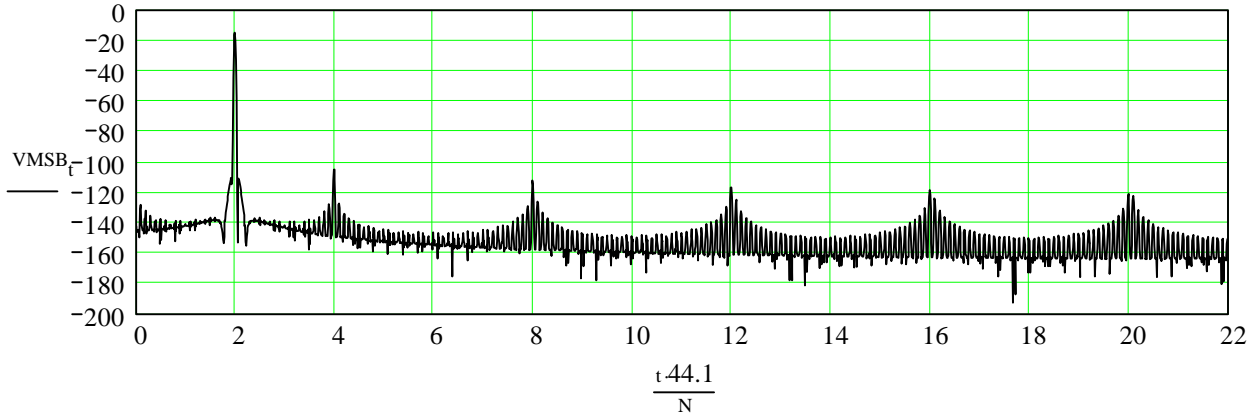
A relative motion between the waves causes no error if the clock "wobble" is a complete full cycle during the time between two sampling clocks. Modulating the clock at exactly the sampling frequency will "complete the cycle" at the sampling times, which are the only points of interest. Modulation by, say, 1KHz above the sampling frequency leaves an undesirable time error due to the motion above the sampling speed.



There are a number of articles written about "Jitter bandwidth". I have seen claims to the effect that jitter above Nyquist does not matter. Others claim that jitter frequencies above sampling causes no harm. Such claims may be true in some context (which I am still unaware of at the time of writing this article). In general, one has to be cautious about ignoring jitter at any frequency.

The case of single or dual tone jitter is not commonly encountered. It is useful for understanding various frequency and amplitude relationships. The case of multiple tone becomes practical when examining electrical coupling of digital data energy to clock circuitry. Let us examine the coupling of MSB digital data energy into the clock circuit. When testing a sine wave, the most significant bit (MSB) follows a square wave pattern at the tone frequency. It thus can be viewed as a decaying sum of odd harmonics.

The impact due to such jitter is shown below for a 2KHz tone.

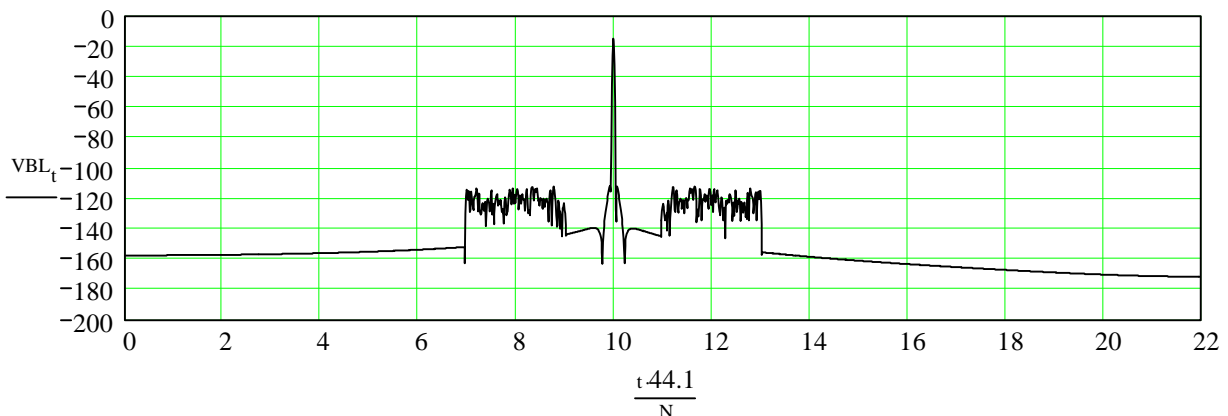


Note: The more informed reader may be puzzled. Clearly, a modulation by the most significant bit (square wave) contains only odd harmonics. Should we not expect sidebands at odd multiples of the fundamental frequency (6, 10, 14, 18KHz...)?

The previous cases of tone modulation were based on a jitter source independent from the tone frequency. In the case of coupling some of the data bits energy to the clock, the clock deviation depends on the signal itself (in a time related manner), thus the unexpected side lobe frequencies.

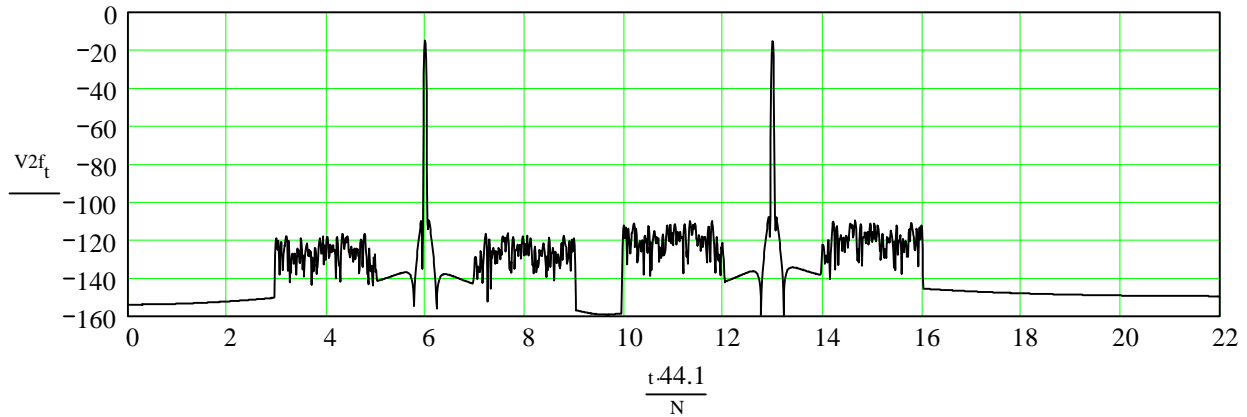
Let us expand the case of signal independent jitter. In practice, we rarely face a pure tone jitter modulation. In fact, much of the jitter problem is random in nature. We will next widen our investigation to a jitter source with the energy spread randomly between 1KHz and 4KHz.

As expected, the "side bands" energy resides in the frequency regions 1KHz to 4KHz away from the tone:

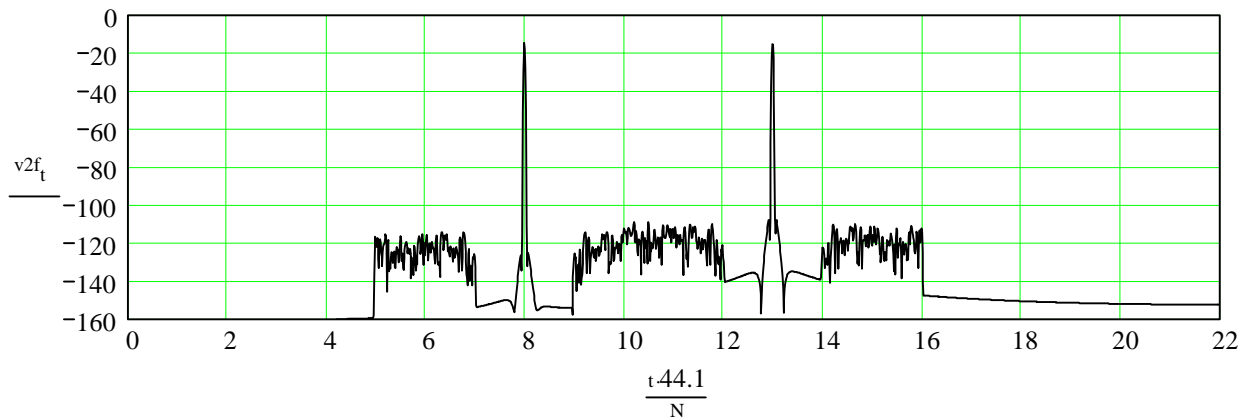


Let us view the impact of the same band limited jitter on a dual tone (6KHz and 13KHz)

Note that the higher frequency tone is associated with higher energy side band

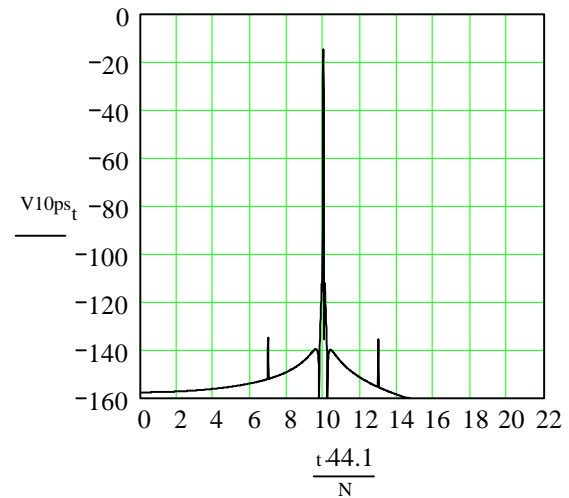
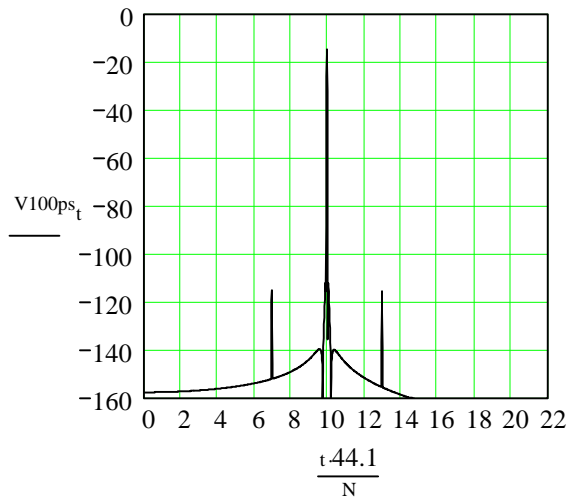


When raising the lower tone to 7.5KHz, we notice that the upper tone sidbands dominate the energy in the overlap region (above 10KHz).



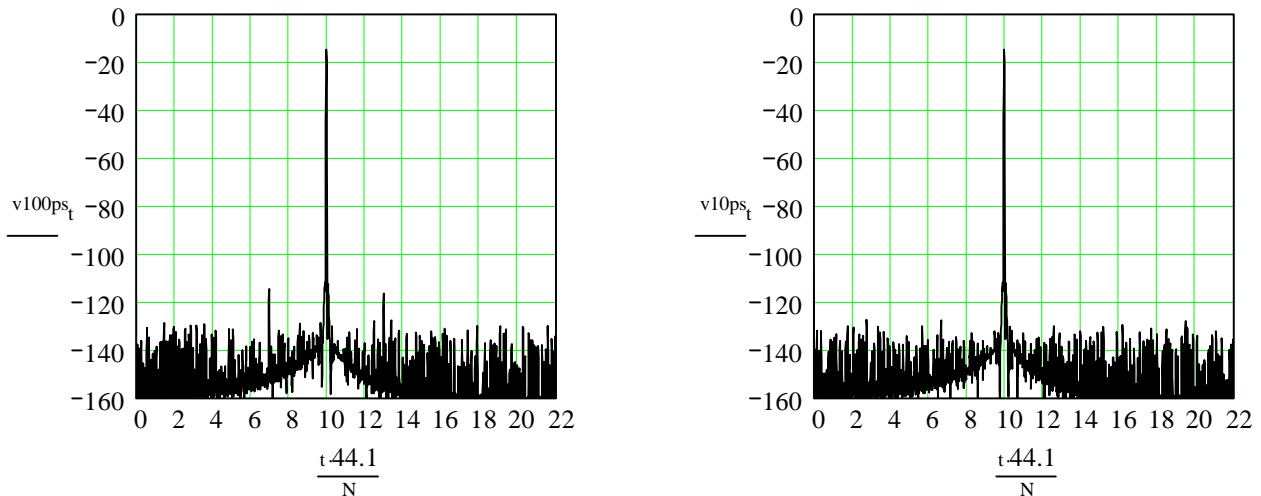
Much has been written about the level of acceptable jitter. The plots below show the sidebands at two jitter amplitudes.

Jitter amplitude of 100pS (left plot) and 10pS (right plot) with 3KHz clock modulation:



How much jitter is acceptable? All the plots so far were based on infinite word length. Jitter side lobes are more noticeable with more bits. The plots below show the impact of same jitter on a 16 bit system. The sidebands with 100pS jitter (left plot) are above the noise floor.

The right plot (10pS) shows that 16 bit truncation dominates the noise floor.



We have seen the impact of sample time errors on analog to digital conversion. The subject of jitter is well covered by literature covering modulation theory (specifically narrow band FM). This article concentrated on PCM coding. Much insight may be gained by classifying jitter in to two categories:

a: Systematic jitter such as narrow band and pure tones tend to generate unwanted energy concentration at frequencies related to the signal and jitter frequencies (the frequency relationships are not always trivial). Systematic jitter occurs mostly due to coupling of periodic signals into the clock circuit.

b: Wideband random jitter tends to modulate (shape) the noise floor in a signal dependent manner (more impact when higher frequency and amplitude signals are present). Random jitter is often due to inherent imperfections of the clock circuitry itself, and to coupling of high speed digital data that is unrelated to signal waveform.

Most coupling mechanisms are due to noisy power supplies, improper ground path and electromagnetic interference. Good design can remove much of the clock jitter and provide clean steady clock to the conversion process.

Hardware related issues

The pervious section points out the undesirable impact of jitter on signal recording and reproduction. This section will point out a number of common sources for jitter in analog to digital and digital to analog converters.

A/D and D/A converters require low jitter clocks. It is easiest to achieve the lowest jitter when using an internal crystal clock, but such a setup is not always practical. Many cases require lock to external clock (multiple devices, stand alone D/A, house sync and more).

Crystals are inherently very low jitter devices, and with proper oscillator circuitry, offer very high immunity to power supply noise and other interference. Crystals can be cut in various ways. Some are made to oscillate in a fundamental frequency mode, and others are made for harmonic oscillation (for higher frequency applications). RLC half bridge, Feedback bridge, Modified Meacham and Pierce oscillator circuits are all good circuits for fixed frequency fundamental oscillations. The integrated circuit single inverter Pierce configuration is the most widely used due to its cost effectiveness.

Clock synchronization requires the ability to vary the frequency over a specified range. The crystal cut is thinner, increasing the series resonance capacity (motional capacity). When coupled with external voltage controlled capacitors (varactors), one may vary the frequency of oscillations over some limited range (up to a few hundred parts per million is possible). Such VCXO circuits (voltage controlled crystal oscillator) are more complex.

A few basic point to keep in mind:

1. Most VCXO's are based on fundamental frequency oscillations (it is very difficult to pull a crystal at an harmonic frequency).
2. It is easier to cut a VCXO crystal for a given a pull range at higher frequencies.
3. Crystals can be made to resonate in series mode or in parallel mode. The author of this article has been getting lower jitter with series resonance circuits.

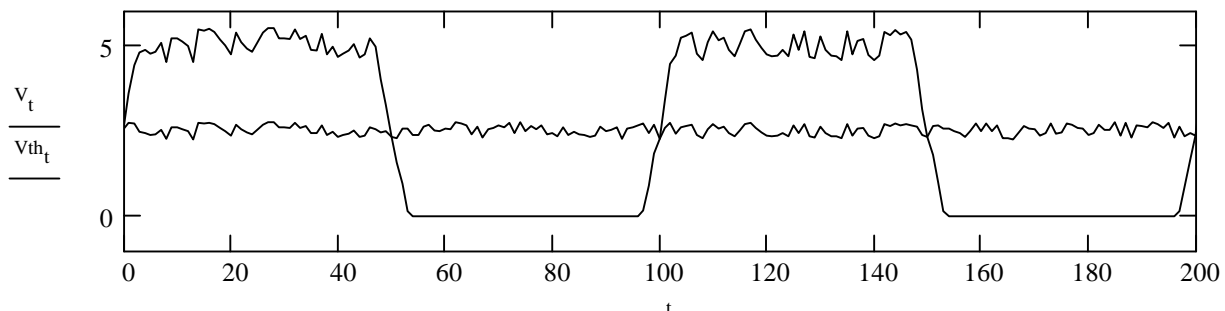
A good "stand alone" crystal oscillator yield very low level of jitter, yet there are two area of concern regarding jitter:

1. Proper clock interfacing to the converter.
2. Noisy or unsteady oscillator control voltage (for the pullable crystal case).

Interfacing the clock

The oscillator clock edge provides the timing of the converter circuit (often through a series of frequency dividers). Sample timing occurs when the clock edge crosses a given logic level. Finite rise (or fall) time, propagation delays and noisy logic threshold can modulate clock timing. This factors are depend on supply voltage thus noisy supplies can cause jitter. The problem is exacerbated when the coupling between the clock supply and the converter supply is poor.

The following plot shows a first order approximation for CMOS gate with 10% power supply noise. The clock $V(t)$ crossing of the threshold $V_{th}(t)$ in noise dependent. An increase in supply voltage speeds up the rise time (earlier crossing) and pulls up the threshold, (later crossing) thus some cancellation occurs when using clock rise time. The converse is true when using clock fall time (slower fall and lower threshold have an additive impact).



The power supply impact on rise/fall time for HCMOS technology is in the order of magnitude of 1nS per volt of noise (about 500pS per volt to the threshold mid point). Other technologies and circuit topologies may yield higher levels of power supply noise immunity. While a faster rise time technology could yield better results in theory, some faster logic families actually exacerbate the problem because the fast transitions become the cause of power supply noise and electromagnetic interference. ECL logic (emitter coupled logic) provide a good solution for clean high speed switching, but at a cost of higher power dissipation. As always, achieving optimal results requires the designer to take into account the various jitter generating mechanisms and find the optimal set of tradeoffs.

The above discussion was focused on clock to converter interface. Any additional circuitry between the clock and the converter (dividers, buffers etc) may add to the jitter with mechanisms described above.

Noisy or unsteady oscillator control voltage

We now proceed to cover the bigger offender: the impact of unwanted voltage variation in the pullable oscillator.

Let us examine a typical VCXO, operating at say 11.2896MHz (256 X sampling rate), with a +/- 150ppm (parts per million) pull range. From a time point of view, each clock period may be pulled by about +/- 13pS, which seem small, yet a 44.1KHz sampling rate requires counting of 256 such clocks periods for each sample. The accumulated time difference thus becomes about 3.4nS. Clearly, such full scale clock modulation is inappropriate, but with +/-3.4nS due to 0-5V control voltage change yields 136pS of peak to peak jitter for 10mV of control voltage fluctuations. Much of the voltage fluctuation is due to the PLL circuitry, though power supply noise becomes a contributing factor with some specific types of PLL. Very little of the fluctuation is due to semiconductor noise voltage.

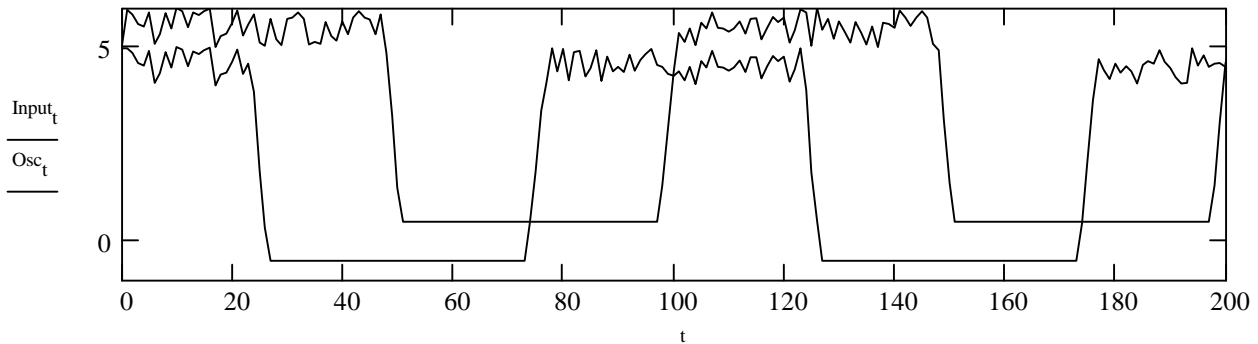
PLL circuits

Achieving lock between the oscillator and the incoming clock requires generation of a control signal, for the purpose of forcing the VCXO to track the time varying input clock. The phase and frequency comparison is commonly done by a phase detector, generating a correction signal for the pullable oscillator. Such negative feedback usually incorporates some filtering between the phase detector and the oscillator.

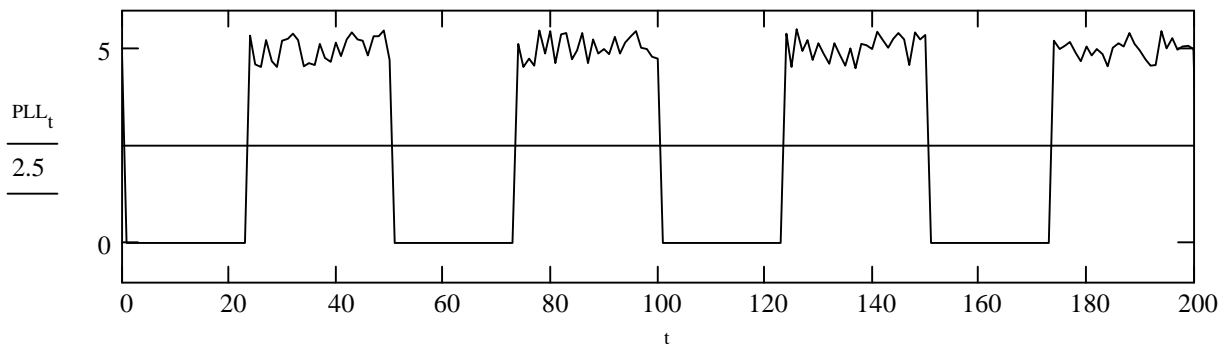
The smoothing (filtering) of the phase detector signal is designed to respond fast enough for tracking the variations in the input clock rate, but slow enough to filter out as much noise as possible. Most circuits utilize a 2 pole filter at 100-1000Hz. Higher order filters are more difficult to operate in the context of a negative feedback loop. Some devices utilize two or more PLL circuits. The improvement due to additional poles is marginal, and the main idea is to lower the filter frequency of the second PLL (noise voltage improvement is proportional to square root of the bandwidth).

Exclusive Or phase detector

When using a simple exclusive-or logic gate as the phase detection circuit, the oscillator locks to the input clock at 90 degree phase shift. The plot below shows the two waveforms with power supply noise imposed on the circuit. The waves are shifted vertically to ease the visual separation.



Next we show the output of the exclusive-or gate. The output signal $PLL(t)$ still contains power supply noise.



The filter does not have to provide perfect smoothing of the square wave into a DC signal. The fundamental and harmonics of the square wave fall at integer multiples of the sampling rate, and jitter modulation is transparent at such frequencies. Similarly, the filter does remove most of the random noise coupled from the supply line. A random noise source of .1V amplitude with flat frequency spectrum from 0 to 20MHz bandwidth yield only 300uV of noise when filtered by a 200Hz low pass filter. The real offender is any significant noise energy below the filter cutoff point. Such energy is mostly due to the coupling of digital logic signals through supply and ground lines.

Another problem associated with the exclusive or gate is the phase shift (90 degrees) between input and output. Some may argue that the fixed 10uS extra time delay is a problem in itself. The author of this article is not convinced of that, but in any case, an input clock variation of only 1ppm at 44.1KHz translates into about 22pS shift in time. Any systematic (or periodic) low frequency content becomes jitter.

The commonly used digital phase frequency detector

Most phase lock loop circuits in audio are based on a digital phase detector utilizing four memory cells (flip flop circuits) and a few logic gates. Such circuits are found in the widely used 74HC4046 integrated circuits. The advantages of such a circuit are:

1. Lock is achieved with zero phase shift (thus no delay or phase modulation).
2. Much freedom from supply noise effects.

Such phase detectors are "idle" most of the time. The (tri state) output is kept at its high impedance mode until called upon to provide a narrow correction pulse. The internal memory is set and reset continuously (set by the input clock and reset by the oscillator). When in lock, a pulse appears at the detector output with duration which is the difference in time between the input and the oscillator rising edges. Such correction is often referred to as "pump up / pump down". A sequence of such pulses (with appropriate polarity), when applied to the smoothing filter, provides the control voltage for the VCXO.

Clearly, while in lock the correction pulses are narrow and the pump is mostly disconnected from the power supply line. Thus the improved supply immunity. (The supply noise impacts the control voltage only during the short correction pulses).

The problem with the familiar 74HC4046 and similar variety, is that phase detector has a "dead zone" (referred to as backlash) of about 2nS of time, when the detector does not respond to the input conditions. When the input frequency moves away from the oscillator frequency, at least 2nS are required before the detector starts generating an output pulse. The outcome is an undesirable "motor boating" effect. Lowering the filter cutoff slows down the modulation but the overall effect can not be filtered out. The backlash and the varying loop gain over the control voltage range make such phase detectors problematic. Special praise is in order for the designers of the latest generation 74HC9046 at Philips Semiconductors. This device operates on the same principles but overcomes the backlash, variable gain and interaction with circuit capacity problems. The 74HC9046 utilizes a switched current charge pump, with no dead zone. The current pump provides an additional benefit of better power supply noise immunity. At Lavry Engineering we will not use the 4046, but we found a couple of slots for the 9046.

Other sources of jitter

While covering of all jitter issues is beyond the scope of any article, the reader must have figured out by now that classical phase lock loop techniques offer little in the way of rejection below the loop filter cutoff frequency. Even with a perfect VCXO, power supply, ground, phase detector and all, incoming jitter residing in the frequency band below the cutoff frequency will be passed on to the VCXO. Achieving and maintaining lock with the classical PLL approach sets limits as to how far one can lower the cutoff frequency.

Much has been said and demonstrated regarding the clock interference caused by the data embedded in digital audio signals. Such interference is at its worst when both audio data and the clock are combined into a composite signal, where transmission bandwidth limitations contribute greatly to jitter. Given any audio tone, one can find some systematic and periodic digital pattern behavior. Thus a large scale (5V) digital periodic bit pattern interference source. The coupling of digital patterns due to audio frequencies below the loop cutoff may remain unfiltered and cause jitter.

CrystalLock (TM) is a better approach

We next describe the basics of the lock circuitry found in Lavry Engineering D/A converters (and some of our A/D converters).

Let us describe the case for an 11.2896MHz VCXO with pull range of +/- 150ppm. We generate the 0-5V control voltage with a 12 bit D/A circuit. The D/A step size is about 1.2mV. Each step changes the clock period by .0064pS, so after 256 clocks we end up with about 1.6pS of jitter over a 44.1KHz clock time period. Let us now run the VCXO with a fixed D/A code for a while, say for 10 seconds, allowing it to completely disregard any variations in input jitter. After 10 seconds have elapsed, we step the D/A up (or down) by one code, varying the oscillator frequency by .073 ppm (300ppm pull range divided by 4096 steps). About 10 seconds later, we may (or may not) step the D/A again but again, only by .073ppm.

Such clock is extremely steady but can we ignore the input data rate fluctuations? Can we "break" the tight relationship between clock rate and input data rate? If the clock is slightly faster than the data rate, are we not going to "run out of data"? If the clock rate is slightly slower than the data rate, do not we have "too much data" for our clock?

The answer is simple: we pre store just enough data in a temporary buffer memory, and clock one sample at a time from the memory with our steady clock. While reading data out of the memory (with the steady clock), new input data continues to fill the memory from its input side. Filling the memory may be done with a lot of jitter yet the output side is ignoring it completely.

The process can not continue indefinitely, because a difference in input and output clock rates will eventually overflow (or underflow) the memory. To avoid the overflow (or underflow), we utilize a processor, responsible to measure and determine if the memory is getting too full or too empty. The processor checks for an average long term trend and makes the small single step clock adjustment as describe above.

A single step adjustment goes a long way: Assume that the D/A is off by one step, then we can operate with "two word deep" memory buffer without overflow for 300 seconds! In practice, the buffer memory is more than two words, because the average input frequency is not constant. An "unrealistic" 100ppm per second input rate drift requires pre storage of about 5 words of data for 1 second D/A stepping, or an 50 word memory for 10 seconds of D/A update rate.

Our DA924 converter operates at an update rate of about 10 - 20 seconds. The very small and infrequent stepping reflect only the very long average drift of incoming data, thus in effect provide a brick wall filter with an "equivalent cutoff frequency" 1000 times lower than the best classical PLL circuits. The control voltage D/A output is highly filtered with a large time constant to smooth the tiny steps, and, though inherently cleaner than CMOS gates, to reject any remaining power supply noise and other interference.