Timing Errors and Jitter

Mike Story[1]

Background

In a sampled (digital) system, samples have to be accurate in level and time. The digital system uses the two bits of information – "the signal was this big at this instant" – and does subsequent maths on it.

In most digital systems, the assumption is that samples are taken on a regular basis, controlled by a clock. They are exactly equally spaced in time. This is reflected in the maths, which is written in terms of z^0, z^{-1}, z^{-2}, etc with z^1 representing a sample period. To do DSP, we do not need all samples evenly spaced – but the maths is best left to someone else if they are not. In any case, modern ADCs are best suited to producing a regular stream of evenly spaced samples.

As in all things where we want perfection – there are sources of error. There are small variations in timing for the reference points at which the samples are supposedly taken, and there are in addition much larger (although less important) errors in the timing of when the samples, or subsequently processed samples, are presented to the outside world. The names for these things are not all well established in DSP literature, so the terms used here are given below. In some literature, the terms "jitter" is used to cover all these effects – but this should be treated with caution.

The errors in the reference point are either (a) random (like noise) or (b) related to the signal. The latter are not often referred to in the literature, but are probably the single most performance limiting aspect of ADCs and DACs as the frequency goes up. Their effect shows up in non-linearity measurements. The third effect (c) is a variation in when a data bit becomes valid, and is mainly of concern to digital designers. It can be a problem for conversion of the digital signal back to analogue. We (*dCS*) refer to these types of timing variations as:

(a) Jitter
(b) **S**ignal **r**elated **t**iming **e**rror (SRTE)
(c) Data jitter

Jitter is often quoted as being the cause of some audible problem when no other explanation is clearly forthcoming. In the author's view, this is not always correct. Jitter certainly causes measurable effects, and these are as audible as the measurement would suggest. It also would appear to cause some effects that are rather hard to explain[2] – but it is probably not the cause of all the problems ascribed to it.

There are a number of methods of measuring jitter. Because we are often looking in the psec region, this is not easy to see on a 'scope, so test methods that measure it by implication are sometimes used. The user should be careful to check that these methods really are measuring jitter, and not something else.

Scale of the Problem

Jitter causes a problem when analogue source material is first converted to digital – because for one reason or another the sample is taken at slightly the wrong instant. It can also cause a problem when a digital signal is turned back to analogue, because the assumption that the signal was "this big for one sample period" breaks down (the DAC gets "this big" right but it gets the sample period wrong). In between, as long as the equipments are synchronous, data jitter is only a problem if it causes the receiving bit of equipment to make an error – and

---

1 with *dCS* Ltd, Mull House, Great Chesterford Court, Great Chesterford, Saffron Walden, UK

2 for example, in one demonstration using the same analogue source material, the same ADC and the same DAC, but changing just the digital interface used between them, one particular backing vocal dramatically changed in prominence

if it causes build up of low frequency jitter such that it cannot be removed by the final DAC. Such a build up can be completely eliminated by the distribution and use of a master clock. For asynchronous links (with asynchronous sample rate converters on the input of the receiving equipment), data jitter causes additional jitter to be processed into the data domain, irrevocably. Asynchronous interfacing should be used with care.

For both (a) and (b) a small timing error δt causes a problem only when a signal is slewing. For an ADC, with an input signal slewing at $\delta v/\delta t$ then the error will be δv. For a DAC, which puts out digital values, the error will be related to the difference between two consecutive samples as: $\delta t/t_s(A_n-A_{n-1})$ where t_s is the sample period. In both cases, the effect of the jitter δt increases as the signal frequency and the signal amplitude increases.

For random jitter (a), a 1kHz sine wave with 1 nsec rms of jitter will cause an amplitude modulated noise signal with rms in the big bits of at –104dB relative to the 1 kHz. The overall rms (taking into account the peaks and troughs of the envelope) is approximately –108dB relative to the 1 kHz sine. See figure 1 for an exaggerated example.

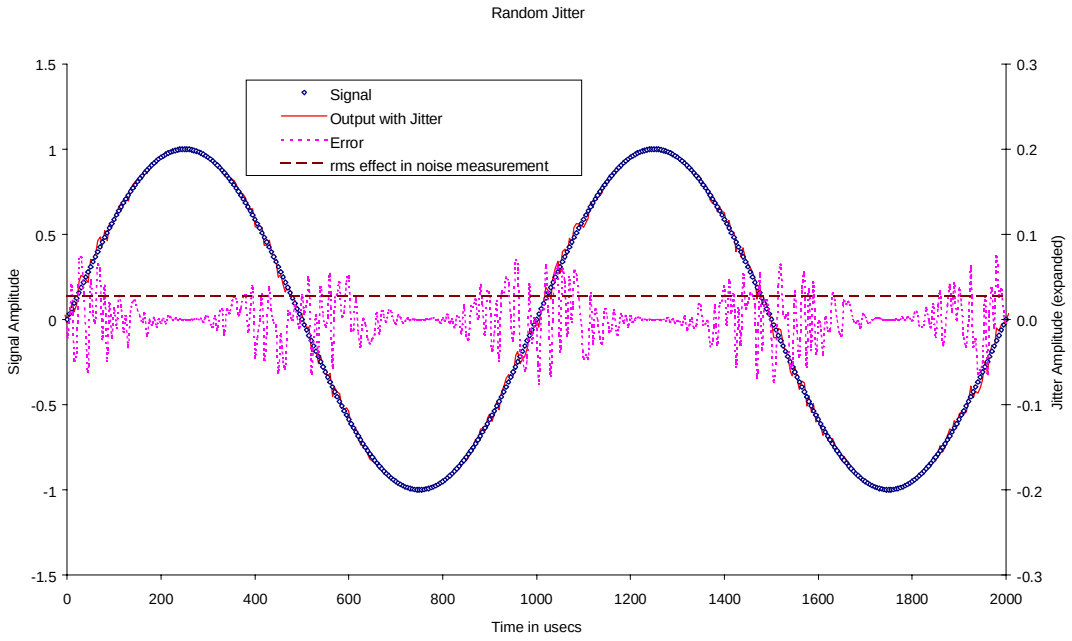

Figure 1

For (b), SRTE, where the timing variation is related to the signal, and synchronous to it, the effect is distortion. Figure 2 shows 1 kHz sine wave with a large SRTE that is proportional to slew rate (the commonest sort). Note that the peak amplitude of the signal is not affected, but that the shape is – the distortion is phase distortion, rather than amplitude distortion.

In reality, SRTE with a peak value of 1 nsec, and whose value is proportional to slew rate, will produce a 2^{nd} harmonic at –110dB relative to a 1 kHz sine, and –90dB relative to a 10 kHz sine. For any circuitry using a sample/hold (residue converters or switched capacitor oversampling converters) this effect is important.

Although not strictly of relevance to audio, for fast circuitry the state of the art for this parameter is about 1 psec, and this limits the accuracy with which a 300 MHz signal can be digitised to about 60dB.

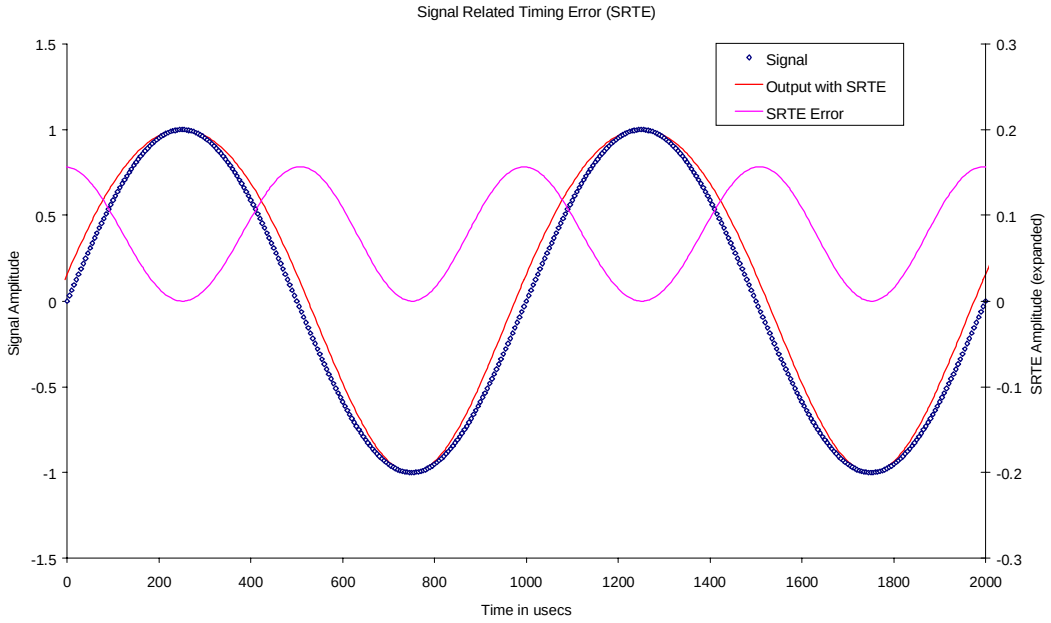

Figure 2 – SRTE proportional to slew rate

Data jitter (c) depends on the hardware used, and is a function of both the source hardware and the interconnect (cabling). The effect can be minimised at source by re-clocking output data just before the output drivers, and making the output drivers high bandwidth. After this, the interconnect bandwidth must not degrade the waveforms. In practice, degradation of waveforms by long cable runs is probably the most important effect. Figure 3 shows the effect of 20m of twisted pair cabling on a double speed AES3 signal for 96 kS/s use, with the top trace the waveform at source prior to balanced pair transmission, and the bottom trace the waveform after the differential receiver at the end of the cable run. The scope is triggered from the transmitted data, and due to cable delays, the transitions are not the same. Data jitter is nearly 5 nsecs p-p.

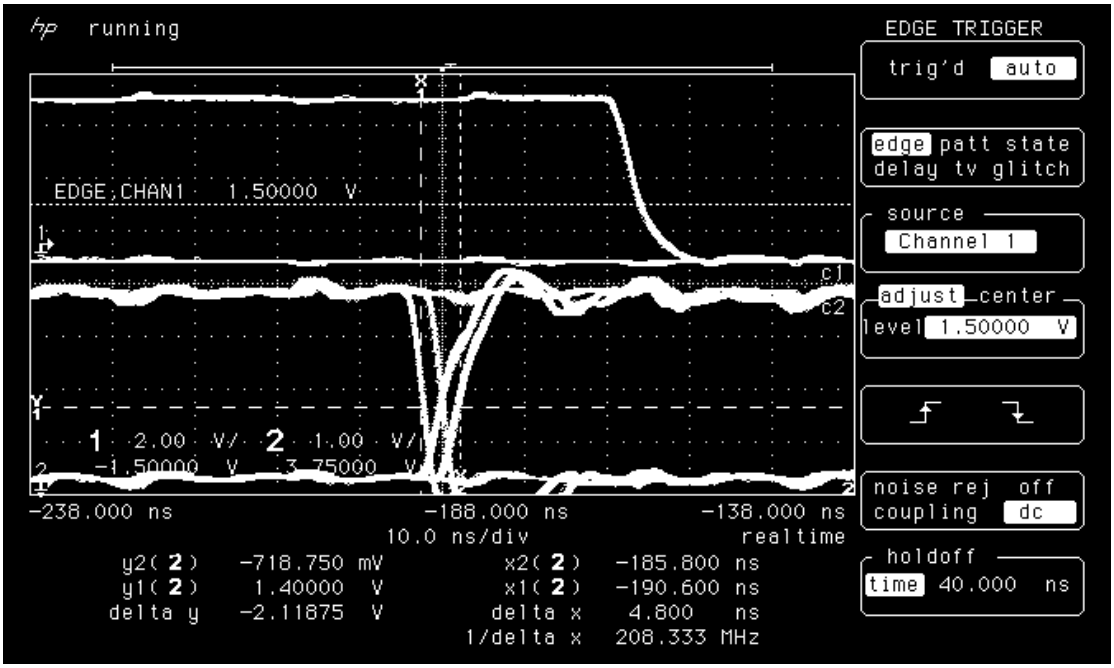

Figure 3 – Data Jitter due to cabling – see text

Where Does Jitter Come From?

(Random) jitter originates in clock circuitry. Crystal based clocks (XCO's, VCXO's) generally have the lowest jitter – but they still have some.

Manufacturers plot the spectrum of crystal oscillators in terms of offset from the nominal frequency, as in figure 4 (solid line). They generally refer to this as the phase noise plot. We can turn this information into jitter as follows[3]:

(a) Assume that half the power in the area of the spectrum away from the nominal frequency is due to phase noise (phase errors, rather than amplitude errors) in the oscillator output.
(b) RMS up the phase errors, as one does for ordinary noise, to get rms radians error (for example, the error sums up to 0.00046 radians rms from 10Hz to 1 MHz offset)
(c) Use the oscillator centre frequency to turn radians into secs (for example, for a 24.576MHz oscillator 1 radian ≡ 6.48 nsecs, so our example has 3 psecs rms jitter from 1Hz to 1 MHz offset.

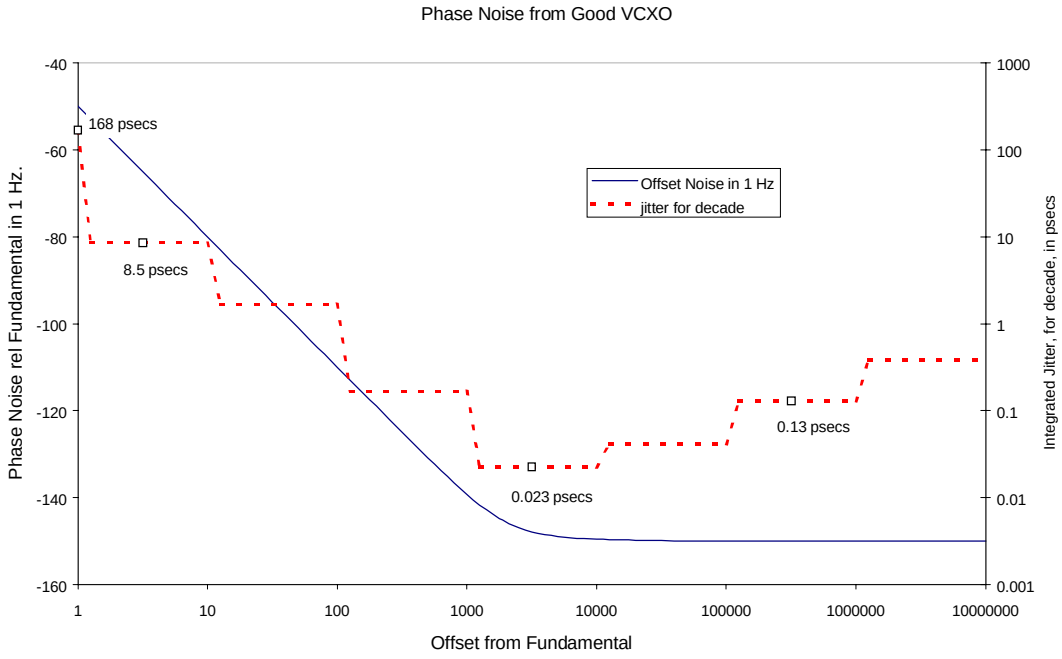

Figure 4

Figure 4 also shows what parts of the phase noise spectrum contribute to jitter. It shows some interesting points:

- The phase noise in the oscillator close to the nominal frequency (close to carrier phase noise) makes a big contribution to the jitter. It is likely that this jitter does not have much perceived effect on audio, because to detect it, our ears would have to have higher Q detection mechanisms than a crystal. In fact, the ears frequency discriminating mechanism has a Q of less than 10, whereas crystals have a Q of 10,000 upwards.
- By 1 kHz, there is very little contribution being made
- However, above this as we approach the "thermal noise floor" of around –150dBc to -160dBc, the effect builds up again by the time noise is integrated up over many MHz. Clock circuitry in digital equipment in inherently quite wide bandwidth, and we must allow for this in our integration.

There are other sources of jitter inside equipment, that may contribute substantially more than the VCXO. Typically, these might be power supply noise causing variation in logic level

[3] See, for example, "Phase-Locked Loop Circuit Design" Dan H.Wolaver, published by Prentice-Hall Inc, 1991. ISBN 0-13-662743-9.

switch points. With a slow edge speed, the variation will cause small timing variations in the apparent switching instant - jitter.

Data Jitter and Clock Recovery

In the absence of a distributed master clock, items in a digital audio chain have to recover their clocks from the data stream they are fed. They will generally do this by locking a local phase locked loop (PLL) to the incoming data stream, and to keep data transfers inside the equipment synchronous, this locked clock is then used internally.

The PLL usually locks to the edges in the incoming data stream, because these are convenient timing points. The incoming edges will have jitter from their controlling oscillator (psecs) and additional data jitter – the latter often of several nsecs. The PLL has to filter this out, which it does by effectively averaging over many incoming edges. The averaging is performed by making the bandwidth of the oscillator control loop small. For example, a control loop bandwidth of 1Hz effectively averages over a period of $1/2\pi$ seconds. For a PLL locking to a 48kS/s AES3 data stream, and looking for sync codes (these occur at 96kHz) averaging is effectively over about 15,000 edges, which will reduce the data jitter by $\sqrt{15,000}$ times (about 124 times) if the spectrum of the jitter is flat.

AES3 at 48 kS/s allows a peak to peak jitter of 80 nsecs on a data edge (it is usually better than this) which is equivalent to about 13 nsecs rms jitter. Reducing this by 124 times (above) will drop it to around 110 psecs rms. In practice, things are better than this:

- the spectrum of the jitter is not flat, but is slightly noise shaped, so the averaging works better than expected, and
- the data jitter is not usually as bad as the spec allows, although long cable runs will make it bad again.

Use of a FIFO to store incoming data works in exactly the same way, by allowing a very low bandwidth in the control loop of the PLL used. It does not eliminate jitter – it is just easier to achieve low control loop bandwidths. Note that for this to offer lower bandwidths than a conventional analogue PLL, and for the FIFO to operate half full, FIFO depths of 64K or more are needed.

If this method of synchronising items in a digital audio chain is used (locking to the previous equipment rather than a distributed master clock), then it becomes important to avoid jitter build up in the area of a few Hz around the centre frequencies of the various oscillators used in the chain.

Data Jitter and Asynchronous Sample Rate Converters

Asynchronous sample rate converters can respond to data jitter and process it into the signal, irrevocably. For this reason they should be used with care, and if they have to be used, should be used with low data jitter sources with short cable runs.

Optical Interfaces

Optical interfaces are sometimes thought to have desirable properties, and occasionally one sees this extended to having some low jitter property.

Optical interfaces certainly do offer very good immunity to electrical interference, and also very wide bandwidth to an optical signal that has been injected into them. However, the problem is usually that:

- the optical power that can be injected into them is limited,
- the bandwidth with which the signal can be injected is limited, or the transmitter is expensive
- and losses at connectors build up rapidly, so that the optical signal reaching the far end is small if more than a few connectors are involved

The signal reaching the far end is usually sufficiently small that the receiver has to carefully control (minimise) its bandwidth to get adequate signal/noise ratio. Minimising bandwidth is just the mechanism that increases data jitter, because it reduces the slew rate associated with a logic transition.

The position can be improved if laser diodes (rather than the more common LEDs) are used to inject signal into the link, because more signal can be injected into the link – but laser diodes are very expensive. Optical links are fine for telecomms applications, where links cover large distances uninterrupted (no connector losses) and expensive transmitters can be used. They are also of value where the high bandwidth of the link makes the cost of a high bandwidth driver (and receiver) economic – for example, some multi-channel applications. For the relatively short links used in two channel audio, the benefits are less clear.

If low jitter is sought, we recommend that optical interfaces should be minimised or avoided.