

Slave to the rhythm

Grimm

www.grimmaudio.com

info@grimmaudio.com

AUDIO

PLL and clock basics

External clocking improves the sound of gear. Sometimes. Sometimes nothing happens, and in some cases the result is downright disappointing. What are the factors involved? This tech note discusses.

Clocks make digital gear tick. Literally: clocks tell digital devices when to perform the next step. However, there are only two places where the stability of that clock affects sound: the A/D converter chip and the D/A converter chip. Only there does the clock get to mix with real life, analog signals because at the moment of conversion clock timing errors convert into amplitude errors. To understand what external clocking does to this clock that the converters see, we need to understand the path followed by the external clock signal.

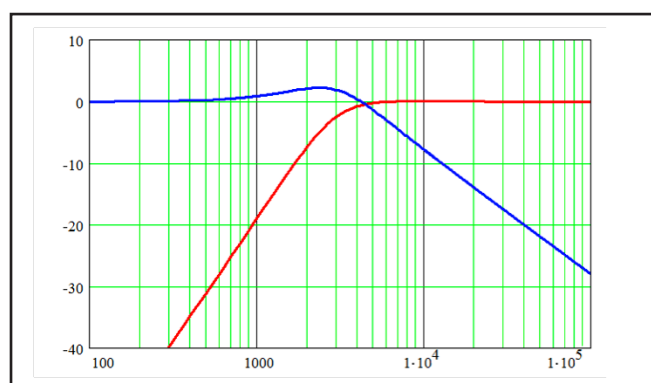
Converter chips need a high frequency clock signal, usually something around 22MHz. It is always made by a local oscillator (anything from a simple RC oscillator in a receiver chip to a crystal oscillator) regardless of whether the unit is operating in master or slave mode. This local oscillator is indispensable: external sync signals may be AES/EBU, a sample rate frequency “word clock” or even a video signal, neither of which are of any direct use to the AD/DA. Instead, the local oscillator is sped up or slowed down to make it run in step with (“locked to”) the sync signal. A system that uses a local oscillator “pulled” in sync with an external signal is called a Phase Locked Loop (PLL). A “phase detector” compares the local clock with the external sync signal and puts out a signal when the local oscillator runs too fast or too slow.

Now, the PLL will track the external sync closely, but not too closely. Unwanted fluctuations (jitter) of the external sync signal are to be ignored but if the local oscillator drifts off it should be pulled back into step. So how does the PLL tell the difference? It can't. It sees only the difference between the two clocks.

All it can do is ignore short term differences (high-frequency jitter), whilst tracking slower fluctuations (low-frequency jitter). After all, a slow change could be due to drifting of either the external sync signal or the local oscillator. Either way, the PLL must get the local oscillator to follow the sync at long last, lest lock be lost.

The cut-off point between “slow” errors and “fast” errors (known as the “PLL bandwidth”) is chosen by the designer, based on an assumption of how stable the local oscillator is compared to the external sync signal. Above the cut-off point, the PLL will not reproduce any errors present in the input, but neither will it be able to correct errors committed by its own local oscillator.

Here's the transfer of a basic PLL, designed to have a cut-off frequency of 4kHz.



Mark that in this graph we are not looking at audio frequencies, but at jitter (or ‘fluctuation’) frequencies. The blue curve shows the attenuation of input jitter. The red curve shows the attenuation of the local oscillator’s jitter. Below the cut-off frequency the external oscillators jitter dominates, above cut-off the local oscillators’. A cut-off frequency at 4 kHz or higher can be found in AES/EBU receivers and general word clock inputs. If he regards the quality of his local clock highly, the designer of a PLL can decide to put the cut-off point much lower, for instance at 10 Hz or even further down. This makes for a ‘slow’ PLL with a very narrow bandwidth.

A quick way of seeing if a PLL is slow or fast is to see how long it takes to achieve lock. Usually, slow PLL’s also take a while to lock. The CC1 takes about 40 seconds to lock and has a 0.1Hz bandwidth. Typical AES/EBU receiver chips lock within a few samples and have a bandwidth of around 10kHz.

If the local clock is very clean, a narrowband PLL is the best choice because all but the lowest-frequency jitter in the external sync is rejected. A converter designed along those lines will sound stellar under all conditions. If the external sync is very clean, a wideband PLL is the best choice because the local oscillator’s own errors will be corrected. This is the case where a good external sync like the CC1 improves a budget converter, or even a pricey one, beyond expectations.

If the designer guesses wrong however, a too-fast PLL might end up forcing an otherwise fine local oscillator to reproduce faithfully every bump and hiccup in the external sync signal. Equipment constructed along these lines sound good in master mode but will only improve in slave mode if the external sync is stabler than the internal oscillator. An unstable external sync actually makes it sound worse.

Alternatively a too-slow PLL might not correct a local oscillator of suboptimal quality. In that case, jitter performance is bad regardless of the quality of the external clock. And here lies the rub: a slow PLL will always make a converter sound the same, but not necessarily good. If a converter is insensitive to external jitter, that alone is no indication that its internal jitter is low. A slow PLL shuts the door to external jitter, but also to any improvement to be had from external clocking with a very stable source.

In short, one cannot expect an external clock to work miracles everytime. If the PLL of the receiving device is slow, the sound quality will be independent of the quality of the external clock, for better or for worse. If the PLL is fast, real improvements can be had.

By example, the graph below shows the result of measurements on a well known DAW converter. The jitter performance, measured at the converter chips’ clock pin, improves substantially at jitter frequencies below 200 Hz when slaved to a CC1.

