



BY GARY GIUST, PHD

Jitter peaking and PLLs

My audiophile brother prizes his tube transistors and vinyl-record collection. It doesn't bother him that every pass of the metal needle on vinyl creates friction that slowly degrades his records' fidelity. Engineers creating clock trees by cascading several PLL (phase-locked-loop) chips may relate to this situation. Jitter peaking with each PLL acts as a "needle" that degrades the timing of the input signal, or "record."

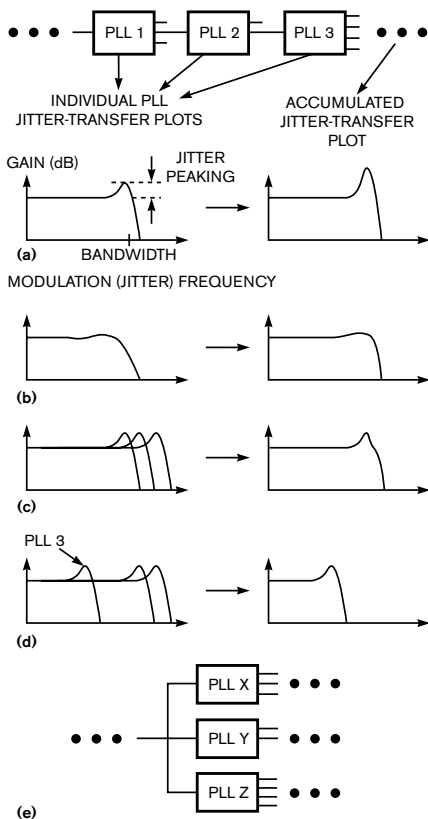


Figure 1 Cascading three identical PLLs amplifies jitter peaking (a). Avoid this problem by selecting overdamped PLLs (b), three different PLLs (c), the lowest bandwidth for the last PLL (d), or an alternative clock tree (e).

As this signal passes through subsequent PLLs, jitter peaking can accumulate to cause instability or timing failures. **Figure 1a** shows a worst-case scenario in which a chosen PLL with significant peaking connects three times in series, causing this peaking to accumulate. The first question to ask in this situation may be surprising: Is jitter peaking really a problem?

Just because jitter peaking exists doesn't mean there is a problem. First, consider the application. Most PLLs exhibit some degree of peaking—typically, 3 dB—and not all applications require the tightest timing margins. At the other extreme, architectures cannot permit jitter peaking to accumulate unbounded. For example, SONET (synchronous-optical network) specifies less-than-0.1-dB peaking. Another way to analyze this issue is to ask: What signals and noise are present in the system? If there is no frequency content in which jitter peaking occurs, then this peaking has little impact on system performance, assuming that the PLL is stable. In other words, if the needle wears a groove in the "vinyl" that you cannot hear or, in my brother's

case, feel, then there is little impact.

What if jitter peaking is a problem? One approach cascades only PLLs having overdamped loops common in applications such as SONET (**Figure 1b**). However, this approach increases cost and limits chips selection. **Figure 1c** shows an approach that avoids using identical PLLs more than once in any clock-tree path. Because each PLL has a different bandwidth, the frequencies at which peaking occurs are staggered, making it difficult for peaking to compound. If multiple vendors act as second sources for the same chip, use each vendor's chip once rather than the same vendor's chip many times, because each company uses different process technologies and design approaches, causing the peaking to occur at different frequencies. When using spread-spectrum clocking to reduce EMI (electromagnetic interference), make sure each PLL's bandwidth is wide enough to pass the spread modulation.

Another approach doesn't restrict how you cascade PLLs, except that the last PLL has the lowest bandwidth of all devices in its path. **Figure 1d** shows that jitter attenuator PLL 3 filters any peaking that accumulates in the tree. However, because PLL 3 is averaging—not tracking—the input-phase error from this peaking, make sure that the PLL can tolerate any worst-case expected phase errors so that it remains locked at all times.

The simplest approach avoids cascading PLLs altogether, in which jitter peaking in any path depends on only one device (**Figure 1e**). When this approach is impossible, request jitter peaking data in decibels and hertz from potential vendors to help understand the impact of cascading their devices and in choosing your available options. **EDN**

Gary Giust, PhD, is a marketing manager at PhaseLink Corp. He also conducts seminars on jitter. Contact him at gary@jittertime.com.