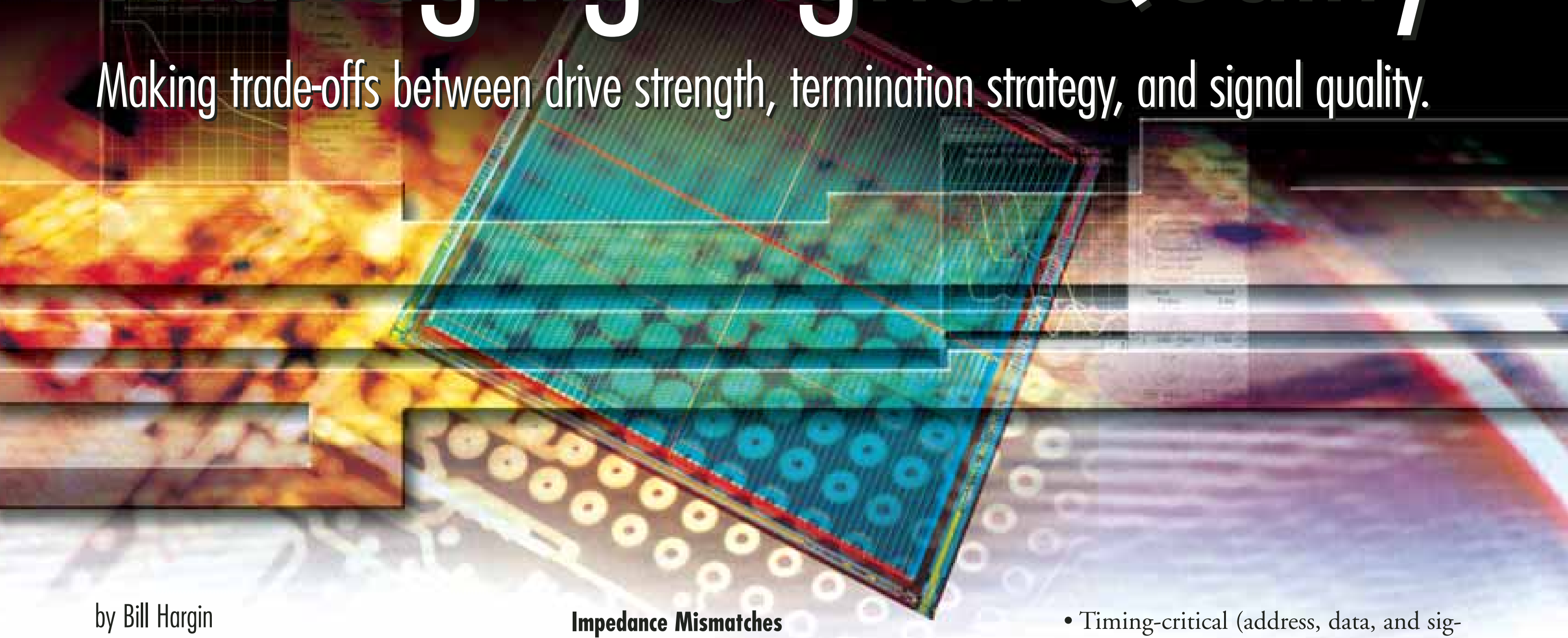
Managing Signal Quality

Making trade-offs between drive strength, termination strategy, and signal quality.



by Bill Hargin
Product Manager, HyperLynx
Mentor Graphics
bill_hargin@mentor.com

Driven by advances in lithography, IC switching speeds continue their progressively faster march. At the same time, escalating clock speeds result in much less forgiving timing margins.

The techniques for managing high-speed effects can be broken into three broad categories, sometimes referred to as "the three Ts":

- Technology – selecting driver technology fast enough to meet your functional needs (but as slow as possible)
- Topology – selecting topologies that meet timing requirements while minimizing the impact of signal reflections
- Termination – managing signal reflections using passive components

Sounds easy, right? The problem is that there are thousands of such choices to be made when designing a PCB, and you must balance these against timing requirements and electromagnetic compliance (EMC).

Impedance Mismatches

Reflections can occur at any impedance discontinuity, including variations in board stackup, trace-width variations, ball-grid-array breakouts, stubs, vias, loads, connector transitions, or large power-plane discontinuities.

The significance of a reflection is impacted by several factors, including the impedance difference, the length over which the impedance difference occurs relative to the overall length of the transmission path, and your technology's tolerance for noise.

Some reflections – if significant enough – cannot be resolved using the three Ts. For this reason, careful pre-layout impedance planning with a tool like HyperLynx's Stackup Planning (shown in Figure 1) is a critical part of a proactive impedance-control process.

The First "T": Triage by Technology

Several strategies exist for dealing with non-ideal routing. The first is to know which nets can accommodate poor routing and which cannot. A "technology triage" strategy works well, dividing nets into those that are:

- Signal integrity-critical (clocks, strobes, and signals that require clean edges)
- Timing-critical (address, data, and signals that can have non-ideal edges but must align with timing requirements)
- Signals with driver edge rates faster than 5 ns

A quick look at the effect of fast driver edge rates is instructive. Figure 2 shows the effect of increasing driver edges on the same 5 inch trace. The 10 ns and 5.0 ns drivers produce clean receiver waveforms. The faster 2.5 ns and 1.0 ns drivers, however, produce reflections and ringing on the yellow and red receiver waveforms.

The Second "T": Topology, Signal Integrity, and Timing

Signal-integrity problems tend to disappear when nets are short relative to how fast they are driven, as reflections settle much more quickly. From the fastest 1.0 ns waveform in Figure 2, the reflections eventually smooth out at a half-inch trace length. Although academically instructive, a health-conscious engineer certainly would not want to specify more than a few carefully planned high-speed nets with a half-inch maximum length requirement.

Sometimes, departures from "good practice" routing can actually be a key to resolving signal-integrity problems. Consider the case of a clock with multiple receivers, each of which is skew-sensitive (the clock must arrive at each receiver at close to the same time). In this case, a daisy-chain route may not be ideal because it delivers the signal to each receiver serially, inherently creating skew.

Here, a superior scheme may be a “star” pattern, in which each receiver (or small subsets of receivers) has its own routing branch. Each receiver can be placed at approximately the same delay length from the driver, and each receiver is considerably more isolated from other receivers than on a daisy chain.

Underscoring the interrelationship and trade-offs between the three Ts, however, star routing introduces several new problems. Multiple branches present the driver IC with a low impedance, requiring it to dynamically sink and source significant current. In practice, you may need to use a stronger driver technology for this topology example, such as a Xilinx® Spartan™-3 LVCMOS33_F_24mA driver instead of a LVCMOS33_F_8mA, as shown in Figure 3.

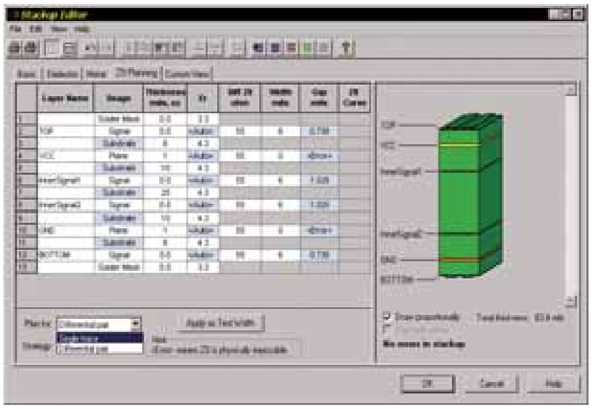


Figure 1 – Careful pre-layout impedance planning using the HyperLynx Stackup Planning tool helps eliminate signal reflections.

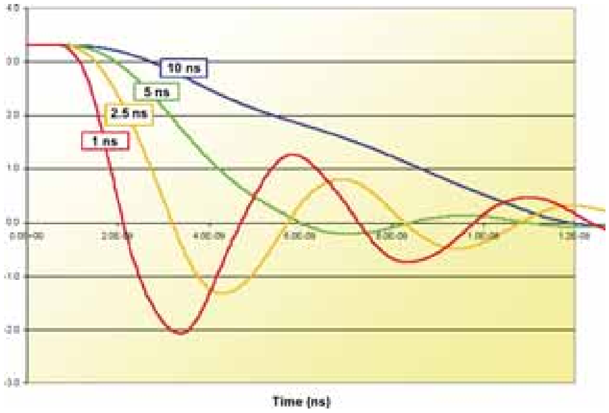


Figure 2 – The effect of driver edge rates on a 5 inch trace. The 10 ns and 5 ns edges look fine, with reflections and ringing effects appearing on the receiver waveforms for the faster 2.5 ns and 1 ns drivers. For sub-nanosecond driver switching speeds, receiver waveforms get progressively worse.

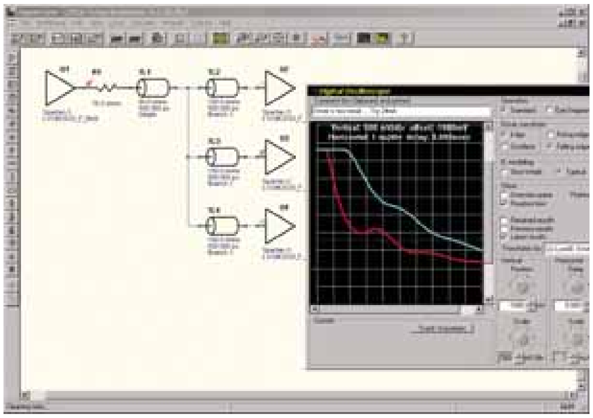


Figure 3 – The HyperLynx oscilloscope and transmission-line schematic show a driver that is too weak for the parallel, low impedance star topology. It would be better in this case to use a stronger 24 mA buffer.

The Third “T”: Termination

As a general rule, any line with an edge rate faster than 5 ns on nets running longer than an inch should be considered a termination candidate. Although reducing costs (see sidebar, “Conserve Board Space with Spartan-3 FPGAs”) is important, the associated signal quality benefits are critical – impacting whether the product works at all. Let’s review some termination strategies for various trace topologies and design requirements.

Termination Types

The classic methods of terminating digital transmission lines are well known. You can terminate the source, the far end, or both; you can employ “distributed” terminations at several locations; or you can use two parallel DC terminating resistors pulled to opposing power supplies to achieve a specific DC bias for Thevenin termination.

Here are some general termination guidelines:

- Source termination is useful in point-to-point/one-directional connections
- Far-end termination is useful in multipoint connections
- Distributed termination can be helpful if you have a plug-in system with variable configuration

Each of these techniques has advantages and disadvantages. Parallel DC termination is the simplest, both from the standpoint of component count (only one, built into Spartan-3 FPGAs) and the choice of value (equal to the line impedance). However, it burns the most power and may unacceptably load the driver IC. AC termination requires an additional component (more expensive, extra board space) and more engineering work (finding the optimal capacitor value), but reduces power consumption.

Series termination creates a voltage plateau that persists until a reflection is received back from the end of the line, so series terminators do not work properly from a timing standpoint unless the receiver ICs are clustered near the end of the net, as shown in Figure 4.

There are several ways to terminate at a junction or star connection. One way is to have a series termination at every driver. This has the advantage of reducing settling time at the receiver while consuming a minimum amount of power. Several conditions must be met for a single-series termination strategy to be effective. Each branch must be close to the same length; otherwise, reflections coming back from each branch are not in sync and end up “bleeding” from branch to branch.

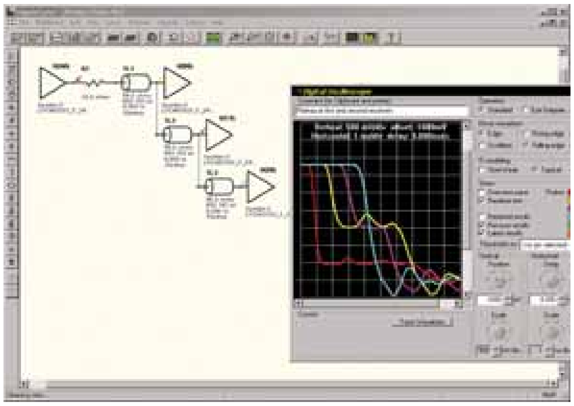


Figure 4 – Series terminators do not work properly from a timing standpoint unless the receiver ICs are clustered near the end of the net, because series termination works by creating a voltage "plateau" that persists until a reflection is received back from the end of the line.

Each branch must also be the same impedance (or close), or it will be impossible to choose an effective single resistor value. If branches are longer than three-quarters of an inch, it makes sense to make their parallel impedances equal to the inbound line impedance from the driver. You can also terminate at the junction itself by changing the trace impedance or by using parallel DC termination – both dampening reflections quickly, and attenuating the signal.

The most appropriate choice depends on network topology and signal direction. For nets that have complex routing patterns, it may be difficult to find a termination scheme that works even in theory. This is where a "what-if" simulation tool like HyperLynx can be an indispensable ally in comparing alternatives.

For a discussion regarding termination component placement, see the "Using Spartan-3 FPGAs to Optimize Termination Component Placement" sidebar.

Conclusion

Spartan-3 devices – with built-in terminators for both single-ended and differential signaling and support for LVTTL, LVCMOS, SSTL, HSTL, GTL, LVDS, and RSDS – are a key enabler for hardware engineers seeking high-speed technology at a reasonable price. But the increased capabilities of modern devices force today's engineers to shoulder the burden of proactively resolving signal integrity, timing, and EMC problems.

With the money you'll save by manufacturing with Spartan-3 FPGAs, consider adding signal integrity analysis software to your toolbox. Several features are important in good analysis software, including the ability to recommend termination strategies and run "what-if" simulations early in the design cycle.

Critical for post-layout analysis are both "interactive" and whole-board "batch" simulation, where violations are flagged and recommendations are made for an entire PCB.

Armed with the latest Xilinx technology, the "three Ts," and the appropriate simulation software, you will not only be ready for today's technology, but ready for what's coming down the road as well. For more information, visit *www.mentor.com/hyperlynx/*.

Using Spartan-3 FPGAs to Optimize Termination Component Placement

Ideally, terminators on a PCB are located at precisely the position they're designed to terminate: exactly at the last receiver IC for parallel termination or exactly at the driver for source termination. Unfortunately, in the real world of dense PCBs, ideal placement is often impossible. Therefore, it's common to find terminators that are located a "stub" length away from their ideal positions. But how long can the stub be before the terminator fails?

Series termination will start to fail when the stub is longer than about 10% of the driver switching time. We can simulate this with a series resistor placed within this guideline and compare the result to a series terminator placed an inch away.

For parallel termination, if the terminator is located upstream from the last receiver on the net, the stub length can be as long as about 15% of the driver switching time.

Fortunately, with Spartan-3 devices, these terminators are located on the FPGA itself, removing the need for placement gyrations and allowing you to focus on using the "three Ts" discussed in this article to address signal quality, timing, and EMC.

Conserve Board Space with Spartan-3 FPGAs

Spartan-3 FPGAs have built-in termination resistors, allowing hardware designers to save significant board space that would have previously been allocated to surface-mount resistors.

Traditional external terminations require two traces for every chip-to-chip connection: a trace for each resistor lead. On-chip termination using Spartan-3 devices cuts this in half, requiring less routing area and possibly reduced layer count.

To get an idea of the benefit, consider a 4 x 6 in, 6-layer PCB with three Spartan-3 FPGAs on it – including a combined routable area of 77 square inches. With previous FPGA technologies, external terminating resistors would consume about 10,000 square mils of board space (40 x 250 mils) each. A board of 4,000 nets (1,333 with termination) would result in a 13.33 square-inch board space savings and a 17% reduction in manufacturing costs.