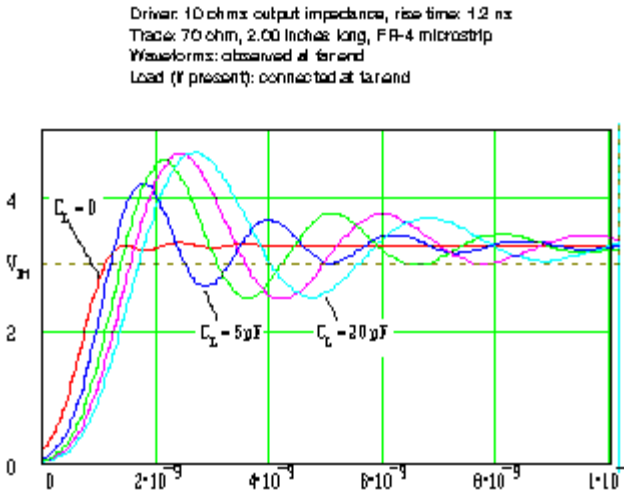
Design Advantage™
We build them for YOU.℠
Comit Systems, Inc. Newsletter
Volume 3 Number 2 Mar. – Apr. 2000

Incorporating Signal Integrity Simulation into the High Speed Board Design Process:

Tool Approaches for Creating & Optimizing a Virtual Prototype.

By Biju Nair

Signal integrity is essential to the proper operation of every high-speed digital product. Ringing, cross talk, ground bounce and power-supply noise etc. can cause high-speed products to fail to operate on the bench or become flaky or unreliable in the field. Today's high speed digital designs based on popular high-performance bus drivers are designed to run at 66, 100, 133-MHz or more, and it takes planning, experience, and more than a little lab work to get digital designs to work at these speeds. Even simple problems, like ringing, can become complex. For example, a typical 3.3-V gate does not need a terminator to drive a 2-inch unloaded trace. The results usually tend to work out all right (see figure. C_L=0).



But, add a single 5-pF load to the end of that same trace, and it will begin to ring, taking an extra 2 ns to settle to within acceptable voltage margins after each transition.

Integrating simulation into the development process and creating and optimizing a virtual prototype can significantly reduce debugging time, thereby shortening the design cycle.

If one can develop a simulation method that has good correlation to the real world, one can use it to simulate PCB trace layouts before one builds them. The computer can be made to check every single trace, with every combination of worst-case loads, which is something one doesn't have the time to do in the lab.

With the right software, a computer can debug a whole month's worth of signal integrity problems without ever committing to a PCB's fabrication cycle. Of course, the key to this whole approach is a good correspondence between the simulator and the real world.

There are generally three tool approaches to signal integrity simulation:

- General-purpose Mathematical calculation environments, like MathCAD, Mathematica, and Matlab.
- Analog simulation tools, like Spice, HSPice, and Pspice.
- Specialized signal integrity simulators, like HyperLynx, Cadence, Mentor/Interconnectix, and View Logic/Quad Design.

General-purpose mathematical calculation tools will allow mixing together of pictures, equations, and explanatory text in the same document. General-purpose math libraries make it possible to simulate the performance of practically any digital transmission structure. Especially good for long-distance communication problems involving frequency-dependent losses (like the skin effect), the flexibility of these tools is invaluable They also simulate a number of different worst-case scenarios, including random topologies, peculiar noise sources, and various flavors of cross talk.

The disadvantage of the general-purpose tool approach is the degree of customization required to model any particular problem. Home- grown customization tends to have not much of a user interface. It may also be extremely awkward to import the trace layout database and work with it.

Traditional analog simulation tools can adequately simulate the most common digital scenarios. All Spice variants have good models for handling simple loss-less, distortion-less transmission lines. These models are appropriate for simulating ordinary printed circuit boards with only a few traces. One nice advantage of working with analog simulation tools is that many of them have a good GUI front-end, with schematic capture capability. That lets you draw a pictorial representation of your circuit, double-check the drawing to make sure it's right, and then submit the completed circuit to the simulator. The pictorial process cuts down on errors.

Spice works exceptionally well when you have good circuit models of both the transmitter and receiver circuits. Its predictions of overshoot and ringing waveforms are surprisingly accurate, given good enough models. The drawbacks to using Spice are that it has no provision for importing a trace layout database, and it has no general way to properly assess cross talk on your layout.

Specialized signal integrity tools attack the trace simulation problem from two new angles. First, they use IBIS models for the driver and receiver. The IBIS models specify, for each driver, a V-I curve to be used in the low state, a V-I curve to be used in the high state, and a finite speed at which the driver transitions from one curve to the other. This arrangement is not perfect, but it's definitely useful. Second, they offer a compact, user-friendly interface, optimized for signal integrity ▶

work. The user interfaces for most signal integrity tools provide for "what-if" operation and "post-processing" operation. In the "what if" mode, a digital engineer can sit down, punch in a topology, and get a quick feel for how a design will perform (Pre Layout Simulation). That is the forte of tools like HyperLynx and the Mentor/Quad Design package called Preview.

In the "post-processing" mode, the software scans the layout database to extract the relevant topological information, picks up an appropriate set of IBIS models from the component database, and then simulates every net, when driven from every possible source (Post Layout Simulation). Ringing in excess of desired specifications (as assigned by the net class) for any net is flagged for the user's attention. Mentor/Interconnectix, Cadence, View Logic, and others all provide this feature.

The Mentor/Interconnectix tool actually adds terminators to fix broken nets, re-simulates them to verify compliance with specs, and then generates change files as needed to show the additional parts. Pretty neat stuff. Of course, the problem with this level of automation is that the raw source information that one types into the system (that is, the IBIS models, the ringing specifications, and the allowed terminator topologies) must be entirely correct for the system to function properly. As with any CAD automation tool, the quality of ones library determines the quality of the final result.

It is important to start simple and keep the simulation closely tied to physical reality. One can start by simulating one NAND gate driving a 10-inch unterminated line, and checking to see if the results match what one sees on the bench. If there are issues, work may be needed on the simulator until results match. Working with logic as fast as 1 ns, models will need to include packaging parasites. IBIS can do this. As confidence with the simulator increases more complex topologies can be tried, with occasional return to the bench for corroboration.

Signal Integrity focused prototyping solutions from different vendors allows one the creative freedom to experiment and investigate signal performance during the design process. One can quickly investigate and constrain design parameters, such as cross talk, interconnect delay, overshoot limits and characteristic impedance for selected nets or signals. And, since Signal Integrity solutions link the logical and physical domains, analysis takes place early in the design process, warning of pitfalls prior to physical implementation. Simulation does not replace final measurement however, since all laws and standards require measurement for verification. Nevertheless, integrating simulation into the development process and creating and optimizing a virtual prototype can significantly reduce debugging time, thereby shortening the design cycle. ■

References: 1. High-Speed Digital Design: A Handbook of Black Magic, by *Howard W Johnson and Martin Graham.* 2. Fast Ethernet: Dawn of A New Network, by *Howard W Johnson.*