BY ERIC SWEETMAN, PHD • VITESSE SEMICONDUCTOR

Measurement-based simulation simplifies analysis of lossy backplanes and cables

A SYSTEMATIC APPROACH USES DSO-BASED TDR AND TDT MEASUREMENTS TO CONSTRUCT MODELS WHOSE SIMULATED RESPONSE DISPLAYS AN UNCANNY RESEMBLANCE TO THE RESPONSE OF THE REAL HARDWARE.

A challenge that often confronts IC designers and support engineers is creating realistic data-channel models and accurately simulating signal transmission through them. Spice models of connectors, vias, and transmission lines work well at lower data rates, but, as data rates approach and exceed 10 Gbps, these models exhibit longer runtimes and become difficult to match to measurements. Behavioral models such as IBIS (I/O-buffer-information specification) execute faster and are evolving toward accurate modeling of transmission lines and nonmonotonic transitions, such as those that incorporate pre-emphasis. All model-based approaches share the problem of matching simulated behavior with that of physical devices, however.

A simulation based entirely on waveforms captured with a DSO (digital sampling oscilloscope) avoids the difficulty of translating physical devices into electrical models (**Figure 1**). You can combine Fourier transforms of time-domain data with S (scattering) parameters derived from TDT (time-domain-transmission) measurements to model signal transmission through complex passive-interconnect systems. Such methods have helped IC designers to simulate the performance of new output drivers over a library of real datapaths and—based on the system's response—to rapidly demonstrate how various parts will perform in a customer's system.

This work began with customer requests for input- and output-port S parameters for asynchronous crosspoint switches in flip-chip BGA packages. Without access to a four-port VNA (vector-network analyzer), applications engineers tried several approaches.

The simplest in concept was to create a TDR (time-domain-reflectometry) test bench in Synopsys HSpice, a version of Spice popular for its W element for transmission-line simulation, and to adjust a multisegment transmission-line model until the simulated TDR provided a good match for the measured TDR. This approach was tedious and resulted in an S_{11} (S-parameter function) that only crudely matched the part's real input behavior.

Applying TDR more directly led to using the FFT (fast Fourier transform) of TDR data to calculate S_{11}. Normalization and de-embedding of board effects were the primary obstacles, but reading previous *EDN* articles led to the use of the current methods (**references 1** and **2**). The normalization process uses the system's TDR response with the BGA part removed from the test board and the solder pads either short or open. In general, the open-pad approach is easier to use on unpopulated boards.

The method captures TDR records from a populated board and a blank board. An FFT routine converts both to the frequency domain. You divide the populated board's complex S_{11} by the unpopulated board's S_{11}. The result is the S_{11} of the BGA part alone. Although Mathcad (www.mathcad.com) easily performed the initial proof-of-principle analysis, later numerical computations used the Scilab (www.scilab.org)

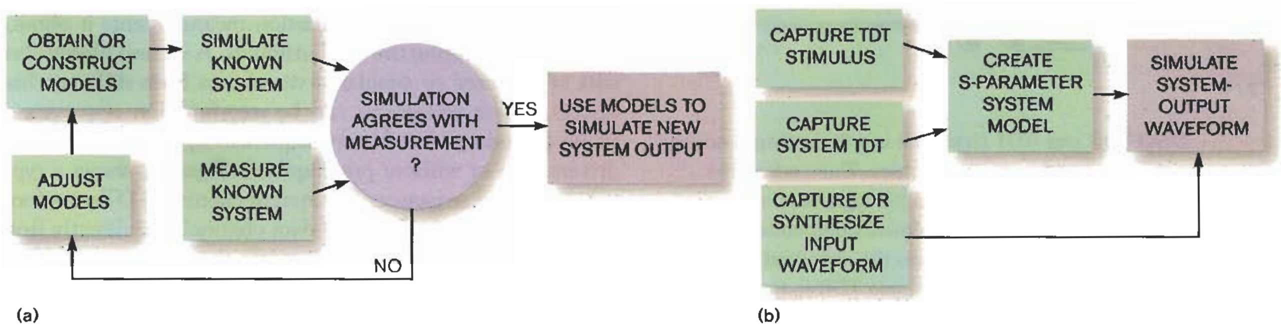


Figure 1 Measurement-based simulation requires acquiring or creating multiple models, such as transceivers, connectors, vias, and striplines. You must adjust the models to match the devices' behavior in a system and then combine them to simulate a new system (a). You can then combine captured or synthesized input waveforms with TDT measurements of the real system to simulate its behavior (b).

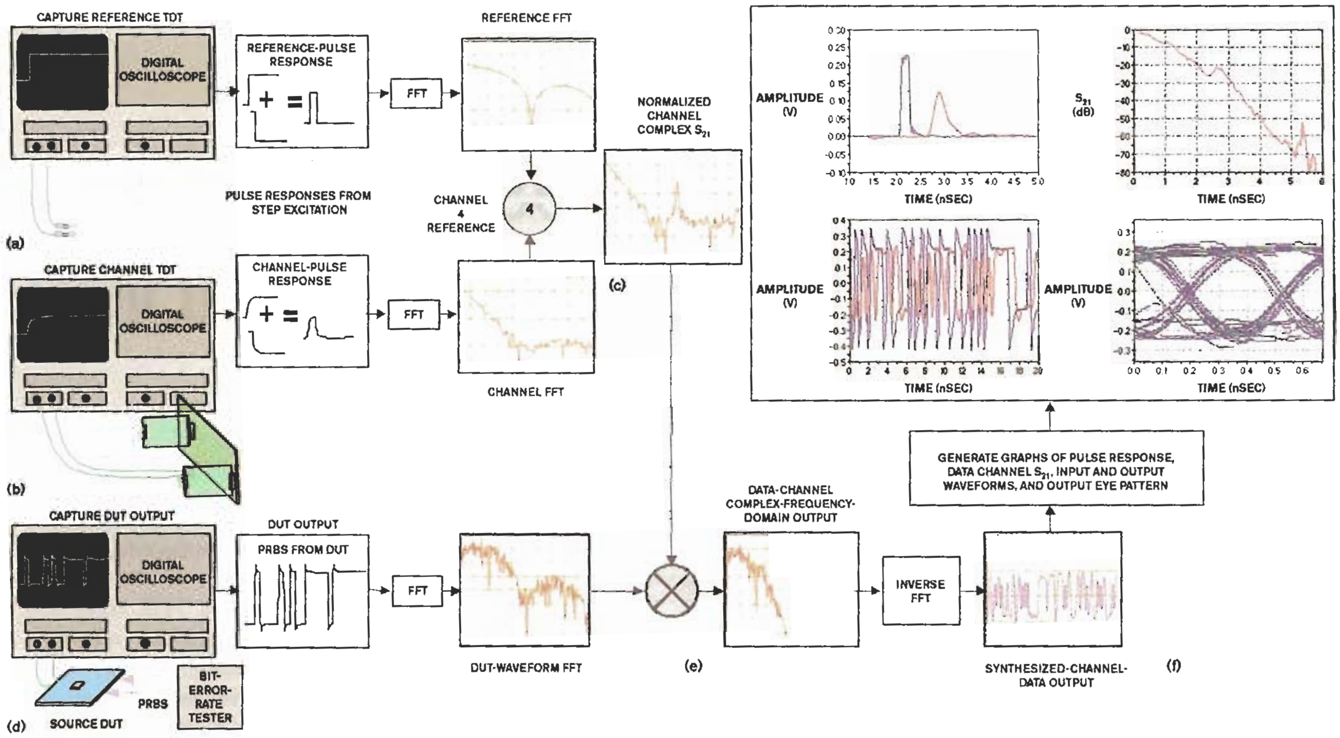


Figure 2 Schematic-data capture and simulation flow follows this path: Start with the TDT-step source (a), the TDT-channel response (b), and a PRBS source (c). Transform the step responses and all waveforms to the frequency domain. Divide these transformed waveforms (d) and multiply them (e) to obtain the system's frequency-domain response. This signal, transformed back to the time domain, appears as a waveform or an eye pattern (f).

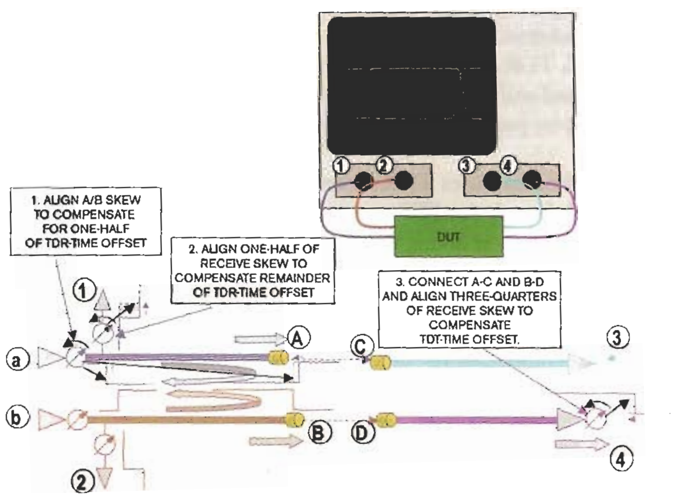


Figure 3 DSO-TDR and -TDT alignment is a three-step process: Observe the TDR signal from open cables. Then, adjust the relative true/complement time to align the signals at cable connectors A and B. Note that this adjustment compensates for only half of the TDR-time mismatch because the observed signals have traversed the cables twice. If you use TDR to generate S_{11} plots, adjust the remaining TDR-time offset at the TDR receivers. After transmission alignment, use short barrel adapters to connect A to B and C to D. Use Channel 4's and Channel 3's skew control to adjust C-3 and D-4.

open-source scientific-software package because of its rich choice of analysis functions.

S parameters can completely describe a linear-data channel. As you can with TDR and S_{11}, you can derive S_{21} from TDT measurements. You can then use these S parameters to model data transmission through backplanes, cables, and other lossy media.

Vitesse (www.vitesse.com) used Scilab to build its simulation environment and based the simulations on TDR and TDT measurements of data channels. This method is flexible because it can capture, simulate, or synthesize stimulus waveforms from step-response waveforms. The simulations transform time-domain data into the frequency domain, in which simple arithmetic operations handle normalization and path effects.

The system stores a library of data channels as TDT-response files along with TDT-normalization measurements it obtains by directly capturing the TDT stimulus. A second library consists of captured or simulated waveforms from different line drivers. These waveforms can be step-function responses or PRBS (pseudorandom-binary-sequence) signals from the same drivers with or without pre-emphasis. **Figure 2** shows a typical simulation sequence. The system captures TDT-waveform data from a lossy backplane's data channel and directly from the TDT-step source and then stores these data in the data-channel library. Both measurements must use the same combination of cables. Both waveforms transform to the complex frequency domain, and their ratio is the data channel's normalized complex-S_{21} response. The channel excitation can be either a captured PRBS waveform or a simulation output from an output-driver model. This time-domain signal trans-

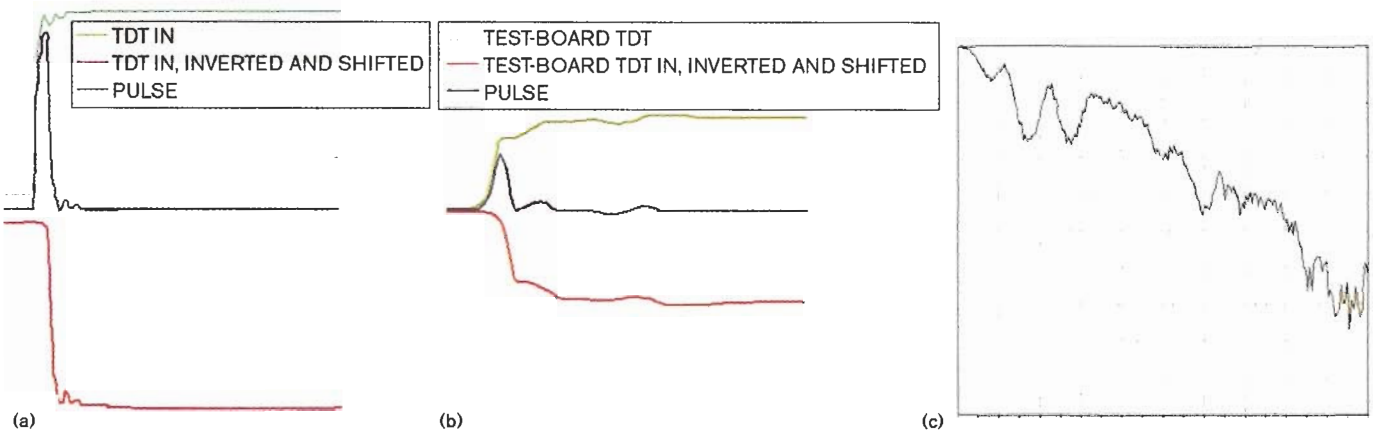


Figure 4 To generate impulse responses and S_{21} from a TDT waveform, start with a reference pulse from the TDT input (a). Next, observe the TDT-output response (b) from the DUT—in this case, an Agilent TDR-test board. Derive the DUT's S_{21} from the input- and output-impulse-response waveforms (c).

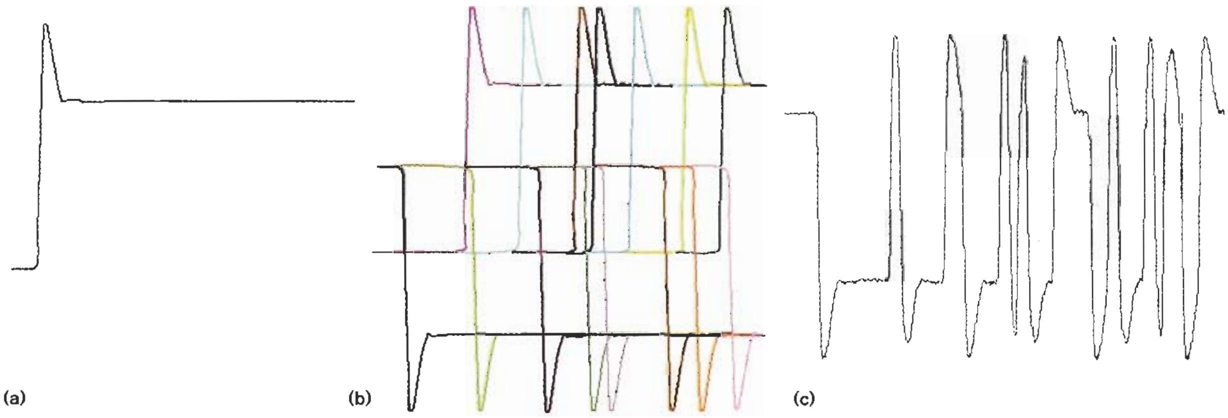


Figure 5 To generate a PRBS from a step response, start with a signal, such as this pre-emphasized step response from a Vitesse asynchronous crosspoint switch (a). Delay multiple copies; add and subtract them to produce a waveform (b). Individual step responses (b) generate the PRBS waveform (c).

forms to the frequency domain, and the normalized channel response multiplies it. Finally, the resulting frequency-domain system response transforms back to the time domain, which displays it as a waveform or an eye pattern.

For TDT-waveform acquisition, you use a DSO with a 35-psec rise-time step source and a 50-GHz-bandwidth input channel. For TDR, the input bandwidth is 20 GHz. A 20-nsec sampling period, 128-waveform averaging, and 4000-point resolution produce good results; averaging for a longer time does little to improve the results.

For differential measurements, you must match the cable pairs as closely as possible and then use the DSO's skew adjustments to better match the step alignment at the DUT's (device under test's) SMA-connector launch point (**Figure 3**). Remembering that the TDR measurement from open cables has double the propagation time of the signal reaching the SMA launch point, set the differential-TDR-step skew adjustment to half the time needed to align the open-cable TDR waveforms. (For TDR acquisition to generate S_{11}, make the remaining half of the skew adjustment on the detector response—in this case, Channel 1 minus Channel 2's skew adjustment.)

Timing alignment is also important for the DSO's receiver channels that capture the TDT waveform. Once you have aligned the launch-cable response, connect the launch cables directly to the receiver cables using SMA barrel adapters and align the TDT-receive-channel skew. Try to achieve 2-psec alignment, but realize that, because the true and complement waveforms are never exactly symmetric, such close alignment can present challenges. In practice, mismatches of 20 psec do not appear to significantly affect the calculated results as long as the imbalance remains constant throughout acquisition of both stimulus and response data.

WAVEFORM CAPTURE

While the launch and receiver cables remain connected, capture a direct-TDT-stimulus waveform, leaving approximately 10% of the time window as a baseline voltage reference before the step (**Figure 4**). You will use this waveform to normalize all subsequent data-channel measurements.

Next, insert the DUT data channel between the input and the output cables and capture a TDT-response waveform, again allowing 10% of the time window for a prestep baseline. Some DSOs don't record timing information. If your instrument doesn't, it is important to keep time-window information for each data file; a simple way is to include the time range in the file name.

Don't try to use an FFT to transform the DSO-captured step-response waveforms directly to the frequency domain. "The worst-possible-case signal to deal with for a DFT (discrete Fourier transform) is the unit-step function" (**Reference 3**). Evaluate the Fourier transform from $-\infty$ to $+\infty$; if the captured data's two endpoints are different, the resulting truncation of a finite time window creates a highly distorted frequency spectrum. The impulse function, which is the derivative of the step function, and the system's TDT response to the step function do not exhibit this problem. Practically, however, the derivative suffers from computational noise. **References 1** and **3** both suggest converting the step response into a finite-pulse response before the DFT. The researcher in **Reference 1** made this transformation by inverting the step response and appending it to the end of the acquired waveform, producing a pulse response whose duration was twice that of the time window. **Reference 3**'s researcher used a pulse excitation, which is the derivative of a step excitation.

A modified version of the approach in **Reference 3** for TDR- or TDT-data delays the waveform by a short time (Δt) and subtracts it from the original signal to create a pulse response with a width of Δt (**Figure 4**). To match the time range of the unshifted signal, the method pads the delayed and subtracted signal at the beginning with the initial level from zero to Δt and truncates at the end by the same amount. The frequency content of the modified data is unchanged, and the value of Δt is not significant. A Δt of less than 90 psec corresponds to the bit width of the fastest Vitesse crosspoint switches, 11 Gbps. The resulting pulse response illustrates the data channel's ISI (intersymbol-interference) effects. This method has the further advantage of creating a 0V baseline, regardless of any measurement offset, such as the one that occurs with ac-coupled signals.

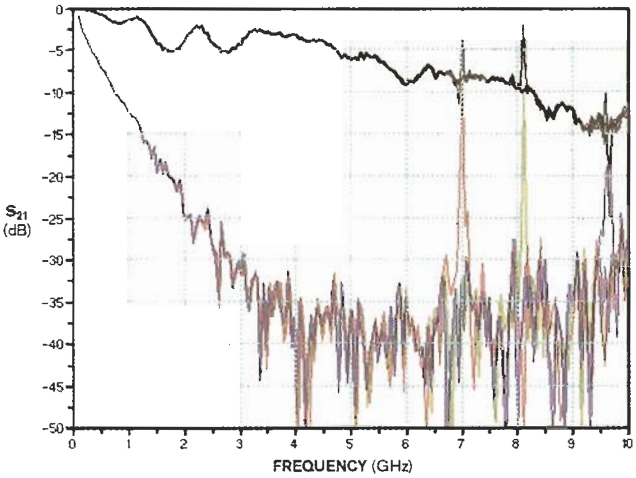


Figure 6 The width of a synthesized pulse affects different data paths differently. These frequency-domain curves are based on TDT data analyzed with three pulse widths of 6.9 GHz at 145, 125, and 105 psec. In the frequency domain, these values result in resonance artifacts. TDT data from 15m of Category 5 cable exhibits these artifacts as peaks at 6.9 GHz (red), 8 GHz (green), and 9.5 GHz (blue), respectively. The upper trace (actually three superimposed traces), from the Agilent TDR-test board, shows artifacts at the same frequencies, but the one at 9.5 GHz is inverted.

With the input and output waveforms converted to pulse functions, calculate S_{21} as the element-wise ratio (Scilab's **./** function) of the complex-pulse-response FFTs to the stimulus FFT. Normalization is automatic. **Figure 4c** plots the magnitude of S_{21} derived from the Agilent (www.agilent.com) test-board's TDT data. In a like manner, you can derive S_{11} from the stimulus and TDR-response FFTs.

OUTPUT-WAVEFORM SIMULATION

Creating reliable S-parameter plots from TDR and TDT data isn't this method's only application. You can also use the method to simulate the effects of a complex path on a data waveform. You can use any waveform as a virtual-signal source, such as the captured PRBS output from a high-speed pattern generator. Because multiplication in the frequency domain is equivalent to convolution in the time domain, you can transform the data waveform into the frequency domain and multiply it, element-wise (**.*** in Scilab) by the datapath's normalized complex S_{21}. The resulting spectrum, when you transform it back into the time domain, is an accurate representation of the source sig-

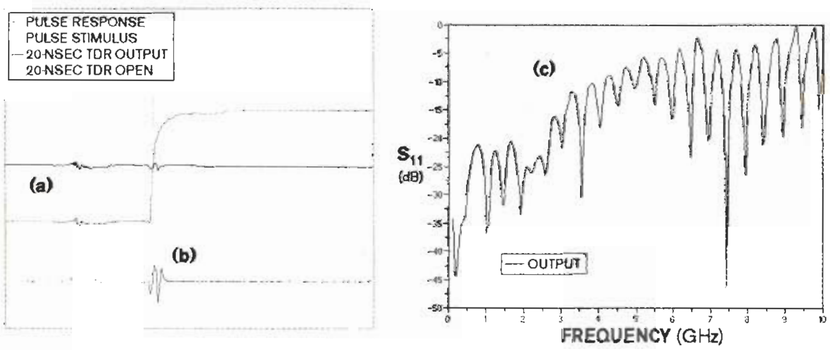
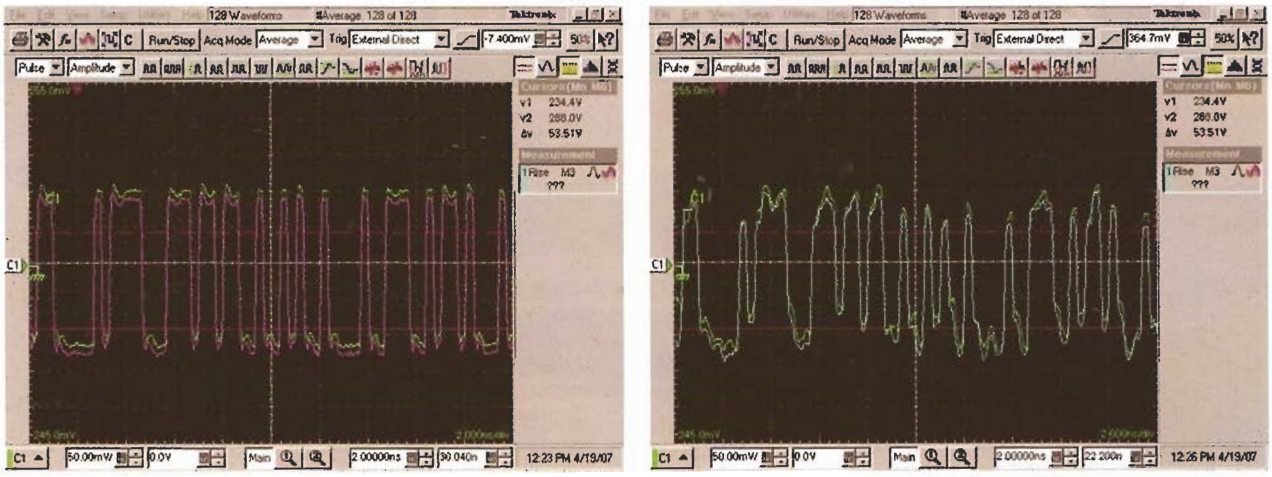


Figure 7 TDR yielded responses at an unpopulated board's output port and at the corresponding port of a populated board of the same design (a); derived impulse and impulse responses (b); and normalized S_{11} with the board effects removed (c).



(a) (b)

Figure 8 Measured test-board results show excellent agreement with simulation: overlay of synthesized (magenta) and measured (green) input waveforms (a) and overlay of simulated (blue) and measured (green) output waveforms (b).

nal propagating through the data channel from which you obtained the TDT.

Vitesse maintains a library of TDT and TDR data from a selection of the backplanes and cables that have passed through the company's laboratory. By using the output from a simulation to stimulate any of the virtual interconnects in its library, the company could, for example, evaluate a new output-preemphasis proposal that engineers had not yet implemented in hardware.

To complement the data-channel library, you might want a library of all likely stimulus waveforms. Unfortunately, though, such a library would be both unnecessarily large and incomplete. A better approach, which works well for Vitesse's asynchronous products, is to use a library of output-driver step responses to synthesize any arbitrary-waveform pattern. You can use such waveforms as simulation stimuli. The advantage of this approach is that it allows you to create a signal of any rate and pattern from a single library file. You synthesize PRBS waveforms in much the same way as the pulsed waveforms you use to evaluate S parameters from TDT data (**Figure 5**).

Capture a step response from a device at a particular output and pre-emphasis setting along with a hierarchy of step responses for each part in the family and store them in an indexed library for later retrieval. To synthesize a PRBS waveform from the part of interest, repeatedly delay and add or subtract—based on the sign and time delay of each PRBS transition—the step response that corresponds to the desired amplitude and pre-emphasis. Ideally, to facilitate Fourier transformation of the PRBS pattern, you should use an even number of component waveforms so that the synthesized waveform starts and ends at the same level. To simulate data-channel behavior, use this synthesized PRBS pattern just as you would use a captured waveform.

TIPS, CAUTIONS, AND LIMITATIONS

This technique is powerful, but you should observe several precautions when you use it. This method models only differential or single-ended transmission paths. In principle, with additional measurements, you could also model even-mode effects or single-ended, multiport configurations.

The step rise time at the DUT input limits the overall bandwidth. Although the bandwidth relates inversely to the time-domain sampling interval, the frequency-domain resolution is inversely proportional to the time window. With 4000 sample points, the combination of a 20-nsec window and a 5-psec sampling interval appears to represent a good compromise.

The source and detector should be proper 50Ω, single-ended, or 100Ω, differential, terminations. The TDT or TDR data ac-

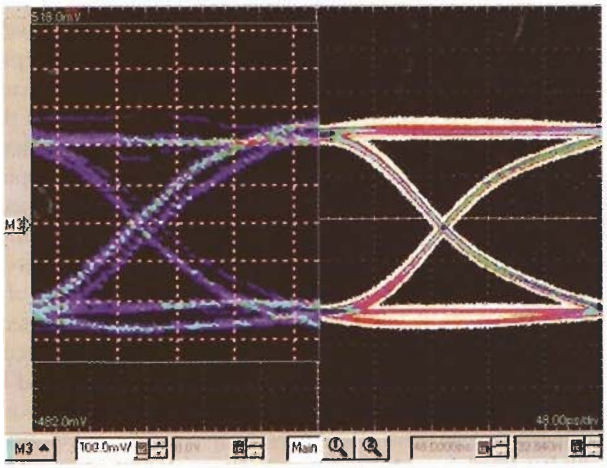


Figure 9 The output of a virtual transmission line (right) driven by a synthesized 4-Gbps PBRS waveform shows excellent agreement with a captured eye (left). These eye plots are for a 2^7-1 PRBS waveform with pre-emphasis from a Vitesse VSC3304HV crosspoint switch driving a 20-in. differential-transmission line.

commodates impedance discontinuities within the DUT, but reflections outside the DUT contaminate the simulation.

The DSO's dynamic range limits the S-parameter noise floor. It is important to use a large fraction of the vertical scale, especially for attenuated signals at the far end of a data channel. Averaging helps, but long averaging can lose time resolution to baseline drift. With one model of DSO, a 128-waveform average appears to be a good compromise.

For evaluating differential datapaths, it is important to drive the system with well-aligned true and complement signals. For differential excitation, the true and complement signals should be mirror images, so that you can use either one in place of a differential measurement. However, you must terminate the outputs of both polarities.

Frequency-domain artifacts can occur at the inverse of the synthesized-impulse width (**Figure 6**). The size of these spurs seems, in part, to relate to triggering offsets that, for long transmission paths, are necessary to place the response waveform on-screen. In practice, these artifacts do not significantly alter simulated-response waveforms.

Some examples better demonstrate the features and accuracy of this simulation method. **Figure 7** shows the original application for measurement-based simulation. You capture a differential-TDR waveform from the output port of an evaluation board on which is mounted a BGA-packaged IC. You capture a second TDR from the same port of a second board that contains no BGA device. You process these step-response waveforms to obtain an S_{11} plot with the board effects normalized out.

Figure 8 shows how Vitesse used single-ended TDT measurements in modeling the transmission properties of **Figure 3**'s Agilent TDR-test board. It uses a pre-emphasized step-response-library file of a Vitesse high-speed crosspoint part to synthesize a PRBS waveform. The method combined the TDT-based S_{21} model with the PRBS stimulus to generate an output waveform. A comparison of the synthesized input and output waveforms with real generated and transmitted waveforms shows a remarkable match.

MORE AT EDN.COM

Go to www.edn.com/ms4273 and click on Feedback Loop to post a comment on this article.

The final example compares a synthesized eye plot with captured eye data (**Figure 9**). The stimulus waveform is a 4-Gbps PRBS signal with pre-emphasis from a Vitesse VSC3004HVEV evaluation board; the datapath is a 20-in. differential stripline.

This measurement-based lossy-transmission-path-simulation method has proved useful to the Vitesse Design Center, in which teams routinely use it for such tasks as TDR-based simulation to characterize new parts' input- and output-return loss. Vitesse has used TDT/TDR to measure the frequency dependence of forward and reverse crosstalk and regularly captures a step-response library from new parts to use for virtual testing of customer backplanes. The center maintains a library of lossy media, such as cables and backplanes, which it uses to optimize new part designs. This virtual testbench continues to find new applications. EDN

REFERENCES

1. Pickerd, John J, "Impulse-response testing lets a single test do the work of thousands," *EDN*, April 27, 1995, pg 95, www.edn.com/archives/1995/042795/09df2.htm.
2. Smolyansky, Dima, "Signal-integrity modeling of gigabit backplanes, cables, and connectors using TDR," *EDN*, July 11, 2002, pg 63, www.edn.com/article/CA231568.
3. Andrews, James R, PhD, "Time-domain spectrum analysis and S-parameter vector network analysis," Application Note AN-16, Picosecond Pulse Labs, May 2004, www.picosecond.com.

AUTHOR'S BIOGRAPHY

Eric Sweetman is an applications engineer for Vitesse crosspoint products, providing data sheets and white papers, evaluating schematics, and performing laboratory tests and simulations. He also guides customer engineers in selecting devices and in achieving standards compliance. Sweetman holds a doctorate in physics from the University of Michigan—Ann Arbor and a bachelor's degree in physics from the Massachusetts Institute of Technology (Cambridge). Outside work, he enjoys digital photography and remote-controlled model airplanes.