Continuous-time equalizers improve high-speed serial links

SANJEEV GUPTA - April 08, 2010

Impedance mismatches on PCBs (printed-circuit boards) and backplanes can cause reflections that impair high-speed chip-to-chip connections. These reflections also create ISI (intersymbol interference), an echo of a previous bit superimposed on the current bit, which causes your system to become unreliable. To overcome the interference, you can employ various signal-processing techniques, including de-emphasis, in which you predistort the signal before transmission. You use an analog filter to make a CTE (continuous-time equalizer) in the transmission path in just a few minutes.

You can best characterize a high-speed digital channel with S (scattering) parameters. S21, an insertion-loss plot, represents the signal attenuation by the channel as a function of frequency. S11, the return-loss plot, represents the reflected signal due to impedance mismatch or discontinuities in the signal path. When a channel path is long, the high-frequency part of the signal suffers severe attenuation (Figure 1).

High-frequency roll-off manifests itself as a rise- and fall-time degradation in the time domain. This degradation limits the data rate you can achieve with the interconnection. You can mitigate this impairment using de-emphasis of the low frequencies at the transmitter or equalization at the receiver.
Reflections also cause echoes in the time domain that appear as phase distortion in the frequency domain, and ISI causes increased jitter and a pattern-dependent variation in the signal amplitude. As a result, the eye diagram remains more closed, making detection at the receiver more prone to error. A CTE mitigates both reflections and ISI.

In preparation for a CTE design, plot and smooth the backplane response with a log-frequency scale.

![Insertion Loss Plot](image1)

Figure 2. In preparation for a CTE design, plot and smooth the backplane response with a log-frequency scale. The insertion-loss plot looks noisy at frequencies greater than 2 GHz. You should perform moving averaging on the insertion-loss plot to remove unwanted noise from the measured data. You can decide on the number of points for the moving average based on the nature of data and to get a smoother response.

You can compare the original insertion-loss response with the moving average.

![Moving Average Plot](image2)

Figure 3. Apply a moving averaging to the insertion loss. The two plots look similar in the passband, but the moving average provides a much smoother response at higher frequency. At 6.25 Gbps, the eye diagram provides insufficient opening for an error-free detection at the receiver.
You can design a pole-zero CTE to achieve optimum performance from a Tyco backplane running at 6.25 Gbps.

Some people refer to the complex frequency plane as the Laplace, or S plane. A simple pole-zero transfer function in the complex frequency plane represents a CTE:

The CTE representation is more complicated in cases such as the HDMI (high-definition multimedia interface).

The PCIe (Peripheral Component Interconnect Express) 3.0 and USB (Universal Serial Bus) 3.0 standards specify the nominal location of the poles and zeros. The exact optimum positions depend on the channel. Fine-tuning the location of pole and zero increases your timing margin for a given interconnect. You should create a mathematical representation to gain more control of the parameters that you can define.

You can represent the equalizer in any math package that supports complex-number mathematics. You can even use Excel for this task if you can figure out a workaround for the software’s lack of support for complex-number math. You can also use models within Agilent’s ADS (Advanced Design System), such as VCVS (voltage-controlled voltage source) PZR, VCVS, and the channel-simulator receiver model, and then write the equations in the ADS data display:

```
Eqr Omega=[0;1e6;10e10]
Eqr zero1=20*log(abs(2*pi*Zero freq)^2*pi*Omega))
Eqr pole1=20*log(abs(2*pi*Pole 1 freq)^2*pi*Omega))
Eqr pole2=20*log(abs(2*pi*Pole 2 freq)^2*pi*Omega))
Eqr Gain=20*log(Gain constant)^2*pi*Pole 2 freq*Pole 1 freq*Zero freq
Eqr Transfer_function=Gain+zero1+pole1+pole2
```

Figure 5 You can also use models within Agilent’s ADS (Advanced Design System), such as VCVS_PZR, VCVS, and the channel-simulator receiver model, and then write the equations in the ADS data display.
You represent the CTE equalizer with a handful of equations on the data-display page. Instead of manually entering these equations, you can use the data-display template in ADS as part of the PCIe DesignGuide. The plot of the transfer-function equation provides the equalizer response in decibels. Plotting the transfer function shows the effect of the pole and zero locations on the equalizer response.

A simple zero location in a transfer function introduces 20 dB of gain per decade; a pole introduces 20 dB of loss per decade. You can visually determine the approximate values of poles and zeros from the frequency-domain response if they have separate frequencies. Draw tangential lines on the parts of the frequency-response curve that have different slopes. The intercepts provide the pole and zero values.

Because a zero in the transfer function introduces gain in the system, you need more poles than zeros to make a system stable. Optimize the locations of poles and zeros to compensate for the channel insertion loss. To design an optimum equalizer, plot the inverse of the pole and zero equalizer response over channel insertion loss with a log-frequency scale.

In this case, the pole and zero locations link to a GUI (graphical-user-interface) slider—the three marker positions, each on linear frequency scales. You use markers 1, 2, and 3 to change the location of poles and zeros until you get the same frequency response as the insertion loss of the channel over the maximum frequency bandwidth. A good match between the two responses provides the optimum design.

In the plot, the blue curve represents the insertion of the Tyco backplane, and the magenta curve shows the inverted response of the pole-zero equalizer in decibels.
You should adjust the sliders until the two curves show good agreement up to the highest frequency of interest, or 2 GHz in this case. The combined insertion loss will be flat to 2 GHz, a large improvement over a nonequalized system, which has an insertion-loss slope starting at 200 MHz.

You could try to write an objective function and use machine optimization, but this manual method provides a visual way of exploring the design and finding out what is possible. For this Tyco backplane, the optimized design provides the zero location at 636 MHz, and the first and second poles are at 1.622 and 6.754 GHz, respectively.
Figure 8 For this Tyco backplane, the optimized design provides the zero location at 636 MHz, and the first and second poles are at 1.622 and 6.754 GHz, respectively.

ADS integrates the equalizer for channel analysis using a VCVS_PZR source.
The channel receiver component also has a CTE component. You plot the eye diagram before and after you insert the equalizer, using the channel simulator in statistical mode. Using this equalizer, the eye diagram shows dramatic improvement in performance (Figure 9). The overall design of this behavioral equalizer takes less than 15 minutes.

Figure 9 You use the channel-simulation schematic with the equalizer to optimize the filter. The channel receiver component also has a CTE component. You plot the eye diagram before and after you insert the equalizer, using the channel simulator in statistical mode. Using this equalizer, the eye diagram shows dramatic improvement in performance.

Figure 10 Using this equalizer, the eye diagram shows dramatic improvement in performance.
In addition to determining the optimum locations of the poles and zeros of a CTE, you can simulate off-the-shelf CTE models within ADS. An off-the-shelf equalizer device comes with tabular data representing the gain- and phase-transfer characteristics as functions of frequency and boost levels. When you use an EDA tool to simulate with these models, you must pay attention to data formatting, data access, and the electrical representation of the device using this data.

Many semiconductor vendors provide CTE data in the CSV (comma-separated-values) file format. It lists frequency as the first column and the real and imaginary components of the transfer function with various boost settings in successive columns. EDA tools cannot use the CSV data, so you have to reformat it. Touchstone, CITI (common instrumentation transfer and interchange), or mdif (measurement-data-interchange format) are acceptable formats for representing a device. Because the standard Touchstone file format supports only one boost setting within a given file and requires a full S-matrix description, it has a limited ability to represent a CTE behavioral model. The CITI file format can easily handle multidimensional data. The Touchstone mdif provides the flexibility to support arbitrary multidimensional data in which blocks of data in a file represent different boost settings. The data is organized in those blocks as a function of frequency, similar to a standard Touchstone file.

High-speed digital designers must use the frequency-domain data to predict eye-diagram performance. While running a time-domain simulation, you must search and interpolate the data with respect to the frequency and boost settings. Simulation environments such as ADS can read all these formats and, using the data-access component, search and interpolate the data as a function of frequency and boost settings. You can access data in MA (linear-magnitude-and-phase-angle), RI (real-and-imaginary), or dBAngle (decibel-magnitude-and-phase-angle) format and use this format with an electrical behavior model representing a CTE.

You can use electrical components such as the equation-based linear SNP (S-parameter/number--f-ports) component, the VCVS, and the system-amplifier model to represent a CTE device, depending on the type of data the device vendor provides. Touchstone SNP components are load-sensitive. You must use correct terminations to accurately replicate the device behavior over various boost settings. If device load conditions are unavailable, you should use a VCVS to represent a CTE device.

Once you convert the data to the required format, you can use the data-access component with an equation-based SNP component or with a VCVS device. You can then use the CTE model in a time-domain environment to predict the eye-diagram and BER (bit-error-rate) performance of the equalizer. You can now do in minutes what used to take days.