Source-Synchronous Testing of Multilane PCI Express and HyperTransport Buses

David C. Keezer
Georgia Institute of Technology

Dany Minier and Patrice Ducharme
IBM Canada

Editor's note:
Source-synchronous I/O buses, such as PCI Express and HyperTransport, are difficult to test with synchronous ATE. This article describes source-synchronous driver and receiver test modules that are tailored for these buses and that have the additional advantage of supporting jitter injection.
—Scott Davidson, Sun Microsystems

THIS ARTICLE PRESENTS a modular approach for testing multigigahertz, multilane digital devices with source-synchronous I/O buses. This approach is suitable for integration with existing ATE and can provide more than 100 independent differential-pair signals. We describe a specific application with 32 lanes of PCI Express, running at 2.5 gigabits per second (Gbps) per lane, and 32 data channels of HyperTransport, at 1.6 Gbps per channel. The differential source-synchronous nature of these buses presents difficulties for traditional (single-ended, synchronous) ATE. We solve these problems by using true-differential driver and receiver test modules tailored for the specific I/O protocols. We satisfy a further requirement for jitter tolerance testing by incorporating a novel digitally synthesized jitter injection technique in the driver modules. The modular nature of our approach permits customization of the test system hardware and optimization for specific DUT test requirements.

Background

Commercially available test systems with high channel counts (more than 512 channels) and moderate speeds (less than 1 gigabit per second) serve routine testing needs. For example, we use an Agilent 93000-P1000 with 896 channels as a host test platform for our project. To address the needs of multi-GHz devices, we enhance the host test system by adding high-speed multiplexing and sampling modules to the device interface board.[1-6] Maintaining tight timing accuracy (less than 50 ps) becomes increasingly difficult and important at higher (multiplexed) data rates.

In the past, researchers used a 200-megabit-per-second (Mbps) automated test system to combine groups of two, four, and eight signals, using simple exclusive-OR gates to synthesize gigabit-per-second stimuli.[2,3] Adding high-speed flip-flops to the test environment provided a way to sample the device-under-test (DUT) outputs at the same rates.[3,4] Researchers also addressed the critical issue of timing calibration.[1-4] Keezer et al. described some refinements to the basic techniques, including an emphasis on modular construction, which facilitates calibration and reuse of high-speed circuits.[5] Wimmers, Sakaitani, and West described an application for testing at 500 MHz and higher.[6]

Researchers later refined these techniques to extend the frequency range to about 5 Gbps.[1,2,5] Keezer, Minier, and Caron addressed the practical issues of achieving (and maintaining) acceptable timing accuracies during production testing. These include automated timing-calibration methods, adjustable I/O voltage levels, DC testing support, and water-cooling the added electronics for thermal stability. The control and testing of jitter effects plays a critical role in ensuring timing accuracy.[7-9]

Device under test

For our project, the DUT is a Northbridge interface chip that controls communications among system

0740-7475/06/$20.00 © 2006 IEEE Copublished by the IEEE CS and the IEEE CASS

memory, processor, external cache, PCI Express bus, and interconnect links with I/O subsystems. This device is part of the chipset used in many of today's PC architectures. It is packaged in a 1,125-pin ceramic ball grid array with 1-mm pitch solder balls. The total number of functional I/Os (not counting power and ground) is 619, many of which carry multi-GHz signals.

Figure 1 shows a top-level view of the DUT I/Os. Our test strategy was to connect the lower-speed memory interface, the front-side bus, and other low-speed control pins directly to the ATE, while testing the PCI Express and HyperTransport buses with customized multiplexing modules.

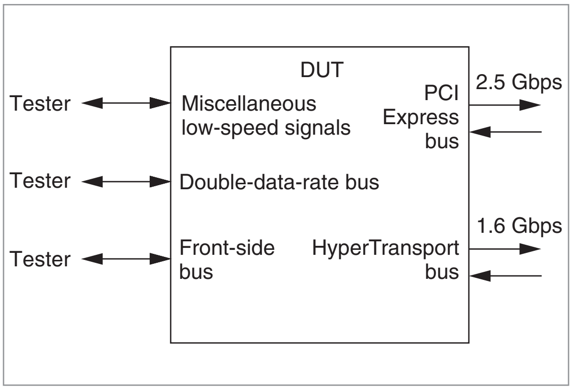


Figure 1. Multidomain device under test.

The HyperTransport links are 16 bits wide (differential) in each direction, operating at 1.6 Gbps per bit. The aggregate data rate is 32 × 1.6 Gbps, or 51.2 Gbps. In the HyperTransport protocol, each group of eight data signals has a dedicated source-synchronous clock channel, so there are 18 differential pairs for each of the two unidirectional ports.

Each PCI Express bus is 16 lanes wide, transmitting or receiving at 2.5 Gbps on each lane. The aggregate data rate is 80 Gbps (32 × 2.5 Gbps). In the PCI Express protocol, the data in each lane is 8b/10b encoded, so that the source-synchronous clock is embedded in each data channel. The input logic circuits must recover this clock at the receiving end, using phase-locking and decoding techniques. This function is not commonly available on most ATE, so we include it in our customized test modules.

Source-synchronous testing requirements and challenges

As emerging high-speed bus technologies become part of new chip designs, new test challenges arise. Traditional ATE is not suitable for such complex and high-speed applications because

- data output from the DUT can be nondeterministic in time,
- data rates can be very high (multi-Gbps), with extreme timing-precision requirements (less than 50 ps), and
- source-synchronous signals require clock recovery driven by the DUT.

Some ATE vendors have addressed these issues by offering new add-in test head cards for their high-end specialized ATE. Others offer external rack-and-stack solutions. Some high-speed cards offer gigabit-range drive and compare for differential signals. Some cards offer loop-back capabilities with jitter injection; others offer clock recovery. Full nondeterministic data features are still not readily available.

For complex DUTs (such as the example application described earlier), using specialized ATE channel cards in the test head requires ATE customization, increasing cost and displacing card slots that would otherwise support high-pin-count standard ATE channels. Testing HyperTransport buses poses the challenge of a multichannel 1.6-Gbps data rate combined with source-synchronous comparison in the receivers. Testing PCI Express pushes the challenge even further by requiring clock recovery combined with a source-synchronous solution to handle potentially nondeterministic data at 2.5 Gbps.

HyperTransport bus overview

In the late 1990s, AMD and API Networks introduced a high-speed interconnect technology called Lightning Data Transfer, which later became the HyperTransport bus. It has been widely adopted and has led to the HyperTransport standard promoted by the HyperTransport Consortium (http://www.hypertransport.org).

HyperTransport uses I/O signal characteristics similar to low-voltage differential signaling for reduced noise and power. This unidirectional bus supports high-bandwidth point-to-point links, reducing I/O contention. The bus data width is scalable from two to 32 differential pairs. Each port has a single control line and a differential clock signal (for every 8 bits of data). Because HyperTransport has a double-data-rate (DDR) topology, it transfers data on both clock edges. Our example DUT application requires a 1.6-Gbps (800-MHz) data rate per channel. The DUT transmits both clock and data from the same clock source to maintain the proper clock-to-data skew. This source-synchronous clocking scheme is necessary to maintain accurate timing at high-gigabit data rates. The HyperTransport Consortium is currently extending the technology to 2.8 Gbps.

PCI Express bus overview

Intel introduced a third-generation I/O interconnect called Arapahoe in 2001 and renamed it PCI Express in 2002. This high-speed serial bus supports chip-to-chip, board-to-board, graphics, and other applications. The bus is scalable from a single lane up to 32 lanes, each running independently at 2.5 Gbps. We summarize its characteristics as follows (http://www.pcisig.com/specifications/pciexpress/base/):

- 100-ohm differential, AC-coupled signaling;
- reduced connection wires;
- serializer-deserializer (SerDes)-based technology;
- increased bandwidth;
- multihierarchy topologies;
- peer-to-peer communication;
- layered structure;
- packetized switched data;
- serial embedded clock interface running at 2.5 Gbps per lane;
- dual simplex point-to-point connection;
- scalable width and bandwidth with several lanes per link (× 1, × 2, × 4, × 8, × 12, × 16, × 32);
- link speed and width lane numbering negotiated between link partners during link power-up;
- 8b/10b data coding; and
- deemphasis equalization.

HyperTransport test difficulties

At 1.6 Gbps, HyperTransport's bit time is only 625 ps, so excessive data or clock jitter is not tolerable. In a standard (synchronous) ATE, the tester's timing subsystem is usually synchronized to a master reference clock using a phase-locked loop. The DUT, however, might have its own embedded reference PLL. As a result, any jitter between the DUT and the tester PLLs will reduce the widths of the data eyes and eventually cause bit errors. On the other hand, if the DUT and the tester are locked to a common timing source, the jitter can still exist but will be correlated from the perspective of the ATE channel that is receiving the DUT data. Thus, in a sense, the ATE jitter can compensate for DUT jitter (rather than add to it). Some ATEs are beginning to offer this ability, at least on a limited basis.

Unfortunately, as ATE size and complexity grow, it becomes increasingly difficult to guarantee phase correlation across all channels. In our modular approach, only the receiving module must be phase locked to the DUT output clock. We do this in HyperTransport by exploiting the available clock signal and using it to drive data reclocking in the receiving module.

PCI Express test difficulties

At 2.5 Gbps, PCI Express's bit time is only 400 ps, even smaller than HyperTransport's. One of this bus's main attributes is its use of SerDes technology, traditionally used in high-speed serial communications. The PCI Express SerDes includes embedded clocks in every lane and therefore requires clock recovery at the receivers. In our test solution, the receiver module logic recovers the clock from data coming from the DUT. Because PCI Express uses 8b/10b encoding, there are sufficient transitions in the data to maintain proper clock recovery. The recovered clock is then used to reclock the received data, so that the tester strobe is now source-synchronous.

After clock recovery in PCI Express, the situation is somewhat similar to the situation we described for HyperTransport. That is, PCI Express's source-synchronous clock has jitter characteristics that are tightly correlated with jitter in the data signal. Therefore, the phase relationship between clock and data is almost constant from one cycle to the next. With the addition of the right amount of timing delay, we can use the recovered clock to reclock the jittery data at the receiver, even when substantial amounts of jitter are present in the original data stream. In our applications, we build this source-synchronous reclocking into the demultiplexing logic.

In addition to synchronizing timing, we must synchronize the expected data's bit pattern with the nondeterministic data from the DUT. PCI Express's SerDes technology automatically inserts special packets called skip order sets and special codes in a nondeterministic fashion. SerDes technology also uses automatic adjustment balancing, called *running disparity*, of the number of 1s and 0s in the transmitted data stream. Furthermore, data packets can be out of order. All these features and requirements mean that the DUT output data can be nondeterministic.[10]

Modular test system

Figure 2 shows a top-level view of the multiplexing test system. It uses multiplexing (driver) modules mounted on the load board to produce high-speed stimuli signals and demultiplexing (receiver) modules to capture DUT output response. The modular approach lets us develop and characterize the driver and receiver electronics independently before mounting them on the load board. The load board provides low-speed connections to the ATE through controlled-impedance transmission lines and pogo pins. Water-cooling the modules is required for heat removal and for maintaining module temperature (to within about 0.5°C) because the DUT is temperature cycled.[1,2] Tight temperature control helps maintain timing accuracy following calibration steps.

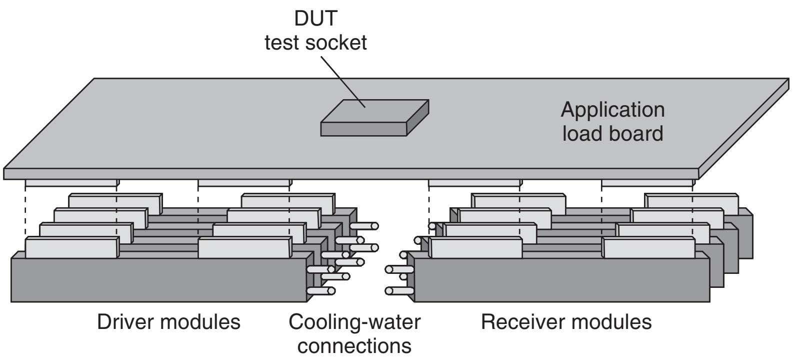


Figure 2. Multiplexing test configuration with driver and receiver modules mounted to the bottom of the load board.[1,2]

Each module consists of a multilayer printed circuit board with components mounted on both sides. The PCB measures approximately 5 cm × 24 cm and has multipin connectors and in some cases individual coaxial connectors (for higher-speed signals). For thermal control, two water-cooled heat sinks are machined to match the profiles of the components on the PCB's top and bottom surfaces. When assembled, the plates and PCB are tightly sandwiched together so that the total thickness of the module is about 2 cm. Thus, as many as 12 modules can be mounted in the space available below the application load board.

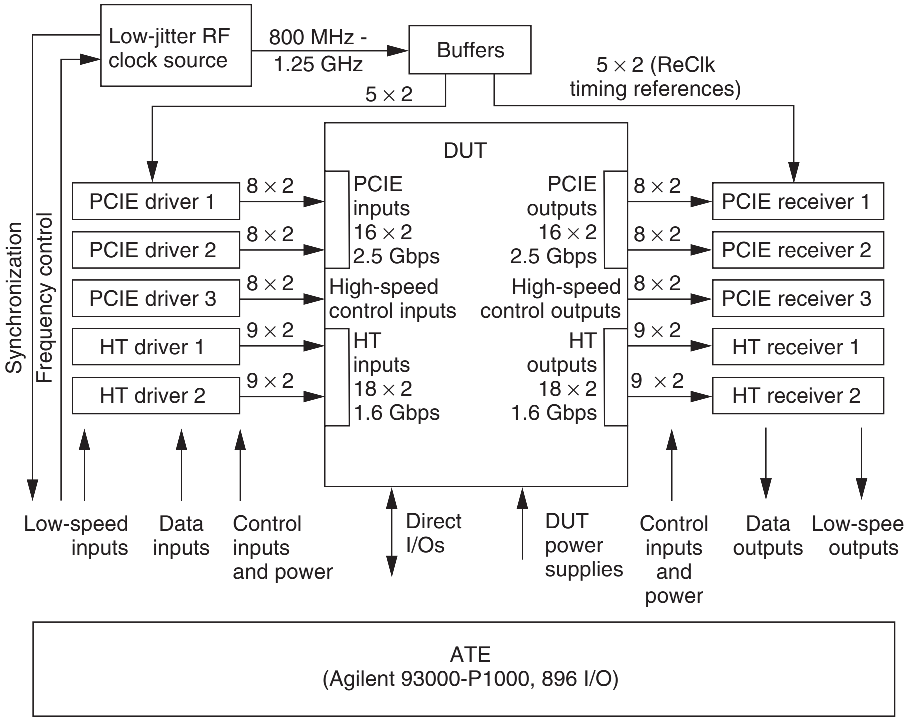


Figure 3. Test configuration for example application (HT: HyperTransport, PCIE: PCI Express).

Figure 3 shows the test configuration for our example application. The DUT requires several multi-GHz input signals and produces several high-speed outputs. The high-speed inputs are synthesized in the driver modules, which are controlled by the ATE using multiple data channels. Various receiver modules sample and demultiplex the high-speed DUT outputs. Low-speed signals connect to and from modules and ATE channels. Relays in the modules switch these signals to permit direct connection of the DUT I/O for DC parametric tests. Other low-speed (less than 1 Gbps) DUT signals connect directly to the ATE.

Controlling and measuring timing of the driver and receiver module signals is critical to the success of the multi-GHz test system. Each module includes embedded logic dedicated to enabling the calibration process.[1,2] These logic blocks require a common timing-

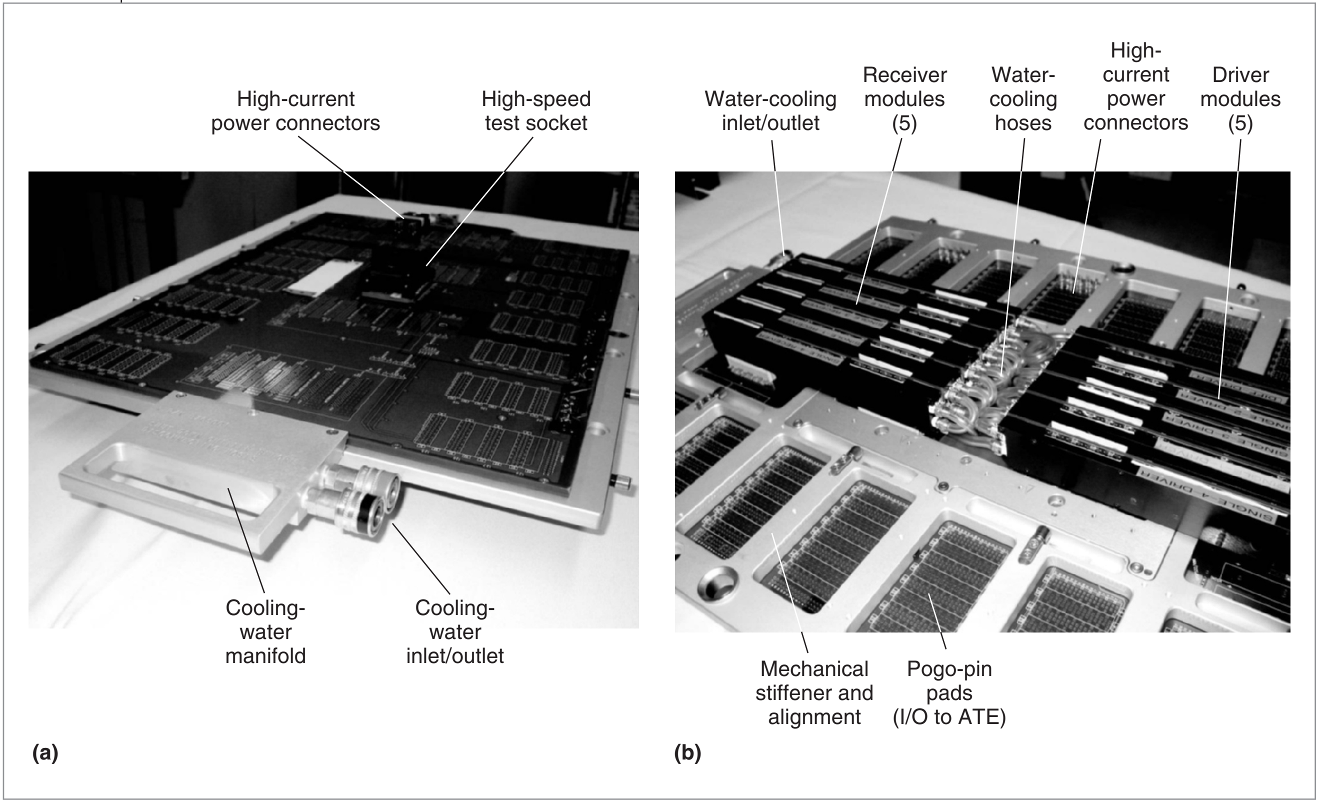

High-current power connectors
High-speed test socket
Cooling-water manifold
Cooling-water inlet/outlet
(a)
Water-cooling inlet/outlet
Receiver modules (5)
Water-cooling hoses
High-current power connectors
Driver modules (5)
Mechanical stiffener and alignment
Pogo-pin pads (I/O to ATE)
(b)

Figure 4. Application load board: top (a) and bottom (b).

reference signal to properly deskew the multi-GHz signals. At system level, we accomplish this by using a low-jitter clock source, phase-locked with the ATE, and distributing it to each module.

Figure 4 shows photographs of the application load board. A high-speed test socket is mounted to the board's top side (Figure 4a). In the foreground are the cooling-water connections and custom-designed distribution manifold. At the far end of the board are large power connectors that handle the high current needed by the driver and receiver modules. Figure 4b shows the bottom view, with all 10 of the required modules. Each module has two black-anodized heat sinks sandwiching the PCB.

The power dissipated by each module ranges from 10 to 25 W (depending on the logic and number of channels in the module). Because logic signals' critical timing depends on the junction temperature, we must control the module temperature to within 0.3°C. We constructed initial prototypes using aluminum heat sinks with forced-air construction.[1] Although these provided the required level of heat removal, the internal junction temperature changed as the DUT was heated or cooled. We could calibrate the multiplexing test system to compensate for this effect, but the amount of time needed to reach thermal equilibrium (about 5 minutes) was unacceptable for production testing. In the final production version, we replaced the forced-air approach with a water-cooled approach, which maintained a test system junction temperature almost independent of the DUT temperature. In about one minute, the case temperature stabilizes to within 0.2°C at about 26°C and maintains this level throughout an extensive test lasting a little more than a minute. Depending on test flow and load board design, module power can remain on between parts, so that the test system requires the one-minute initial stabilization time only at the beginning of a batch.

Driver modules

To exploit the modularity of our approach, we constructed several different driver modules. Table 1 lists some examples. Each is designed to optimize ATE resources while meeting specific DUT test requirements. The modules implement various types of multiplexing schemes, ranging from 2:1 XOR-based multiplexers[1-9] to 32:1 parallel-to-serial converters. The simpler arrangements (XOR multiplexers) provide maximum data rates, which are limited by the ATE's timing accuracy to about 2 Gbps. Higher performance (3.2 and 5.0 Gbps) requires reclocking the multiplexed data using a low-jitter clock source. Here, we review earlier designs[1,2] and then introduce Driver9-PCIE and Driver10-HT, which are optimized for the PCIE and HyperTransport applications.

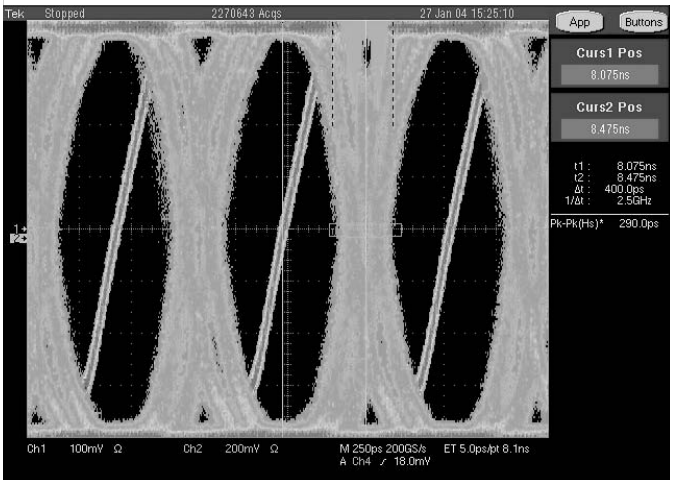
Driver3 uses an external low-jitter clock source to reclock multiplexed data. This reclock signal generator is phase-locked with the ATE system's master clock so that the multiplexed signals produced by the driver are also locked to the ATE. Figure 5 illustrates the reclocking operation, in which both the multiplexed data eye and the reclocking signals are displayed for three data bit periods at 1.25 Gbps.[2] The data eyes exhibit similar jitter characteristics as in previous designs. The three diagonal lines in the figure are the rising edges of three reclock cycles (the data patterns obscure the falling edges). Measured separately, this clock has random jitter of about 1.5 ps root-mean-square (rms), or about 20 ps peak to peak. The data eye opening at this speed provides plenty of timing margin for reclocking.

Figure 6a illustrates the basic logic implemented for each high-speed signal in Driver9-PCIE. This module multiplexes groups of four signals from the ATE. With an ATE base frequency of 1 Gbps, this permits generation of multi-GHz signals in the driver modules. Each high-speed channel also includes a variable-level output buffer with adjustable amplitude and DC offsets, embedded calibration logic, and high-performance relays. To achieve the desired time delay at the DUT input, the calibration logic performs timing alignment (deskew) of the signals produced by the multiplexer. A low-jitter (about 1.3 ps rms) calibration reference signal is distributed throughout the system and provided as input to each module. A digitally programmable delay circuit adjusts the phase of each multiplexed channel with a 10-ps timing resolution. A multistage, iterative calibration procedure obtains timing accuracy limited by the 10-ps resolution.[1,2]

We use this calibration approach to measure the nonlinear timing characteristics of each programmable delay circuit (usually at about 1-ps resolution). The digital codes are then adjusted to obtain timing delays with accuracy of about ±10 ps (not including jitter effects). We also measure fixed delay values for each high-speed channel within the modules and for each transmission line between the modules and the DUT I/O. Again, these are characterized to a precision of about 1 ps, so that the accumulated timing errors are in the 10-ps range.

Table 1. Example driver modules.

Design	Function	No. of channels	Fmax (Gbps)
Driver1	4:1 XOR mux	8 differential	2.0
Driver2	4:1 XOR mux	12 differential	2.0
Driver3	4:1 ReClk mux	12 differential	2.5
Driver4	2:1 XOR mux	24 single-ended	1.8
Driver5	8:1 ReClk mux	4 differential	3.6
Driver6	16:1 ReClk mux	2 differential	5.0
Driver9-PCIE	4:1 ReClk mux	8 differential	3.2
Driver10-HT	2:1 ReClk mux	9 differential	2.0


Tek Stopped 2270643 Acqs 27 Jan 04 15:25:10

Curs1 Pos 8.075ns

Curs2 Pos 8.475ns

t1 : 8.075ns
t2 : 8.475ns
Δt : 400.0ps
1/Δt : 2.5GHz

Pk-Pk(Hs)* 290.0ps

Ch1 100mV Ω Ch2 200mV Ω M 250ps 200GS/s ET 5.0ps/pt 8.1ns
A Ch4 ⨍ 18.0mV

Figure 5. Driver3's internal reclocking.

The relays contained in the driver module permit switching of ATE channels directly to the DUT inputs (bypassing the multiplexing logic). This allows the ATE's DC parametric units to function in their normal fashion to measure characteristics such as input leakage, continuity, and input voltage sensitivity (VIL/VIH).

As Figure 6b shows, the Driver9-PCIE module produces the ReClk* signal by dynamically phase-shifting the low-jitter ReClk reference signal under control of the host ATE. Here, another programmable delay circuit under digital control of the ATE modulates the low-jitter ReClk signal's phase. This approach allows the ReClk* phase to be updated at the ATE test vector rate (up to 1 billion times per second). To synthesize additional jitter on the ReClk* signal, we program a prescribed sequence of digital control values (usually pseudorandom) into the ATE test vectors, resulting in added jitter in the multiplexed data signals. By chang-

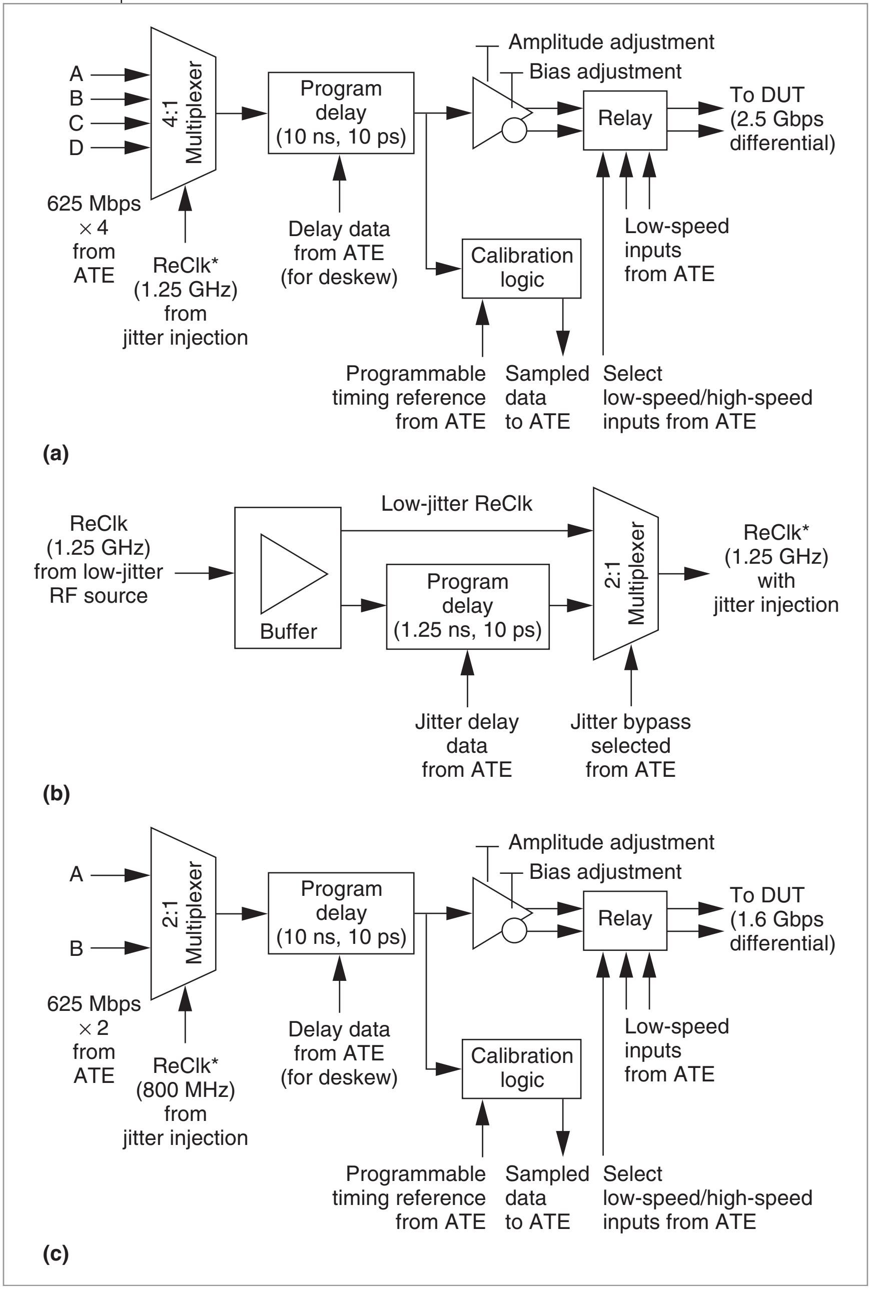

A
B
C
D
4:1 Multiplexer
625 Mbps × 4 from ATE
ReClk* (1.25 GHz) from jitter injection
Program delay (10 ns, 10 ps)
Delay data from ATE (for deskew)
Amplitude adjustment
Bias adjustment
Relay
To DUT (2.5 Gbps differential)
Low-speed inputs from ATE
Calibration logic
Programmable timing reference from ATE
Sampled data to ATE
Select low-speed/high-speed inputs from ATE
(a)
ReClk (1.25 GHz) from low-jitter RF source
Buffer
Low-jitter ReClk
Program delay (1.25 ns, 10 ps)
2:1 Multiplexer
ReClk* (1.25 GHz) with jitter injection
Jitter delay data from ATE
Jitter bypass selected from ATE
(b)
A
B
2:1 Multiplexer
625 Mbps × 2 from ATE
ReClk* (800 MHz) from jitter injection
Program delay (10 ns, 10 ps)
Delay data from ATE (for deskew)
Amplitude adjustment
Bias adjustment
Relay
To DUT (1.6 Gbps differential)
Low-speed inputs from ATE
Calibration logic
Programmable timing reference from ATE
Sampled data to ATE
Select low-speed/high-speed inputs from ATE
(c)

Figure 6. Logic implemented by each of the eight Driver9-PCIE channels (a); generation of ReClk* with injected jitter from the low-jitter ReClk reference signal (b); logic implemented by each of the eight Driver10-HT channels in each module (c).

ing the digital program, we can synthesize a wide variety of jitter effects, such as pseudo-Gaussian, sinusoidal, deterministic, or various combinations of these.

Figure 6c shows the logic implemented by each of the eight data channels in Driver10-HT. This logic is similar to that of the Driver9-PCIE, except that the Driver10-HT includes a 2:1 multiplexer. The ATE input data rates to the multiplexer are 800 Mbps, and the resulting (multiplexed) output data rates are 1.6 Gbps.

Figure 7 shows a photograph of the prototype Driver9-PCIE module. Some module features are visible, including the 18 coaxial connectors (for the eight differential data channels and differential clock). An array of high-speed relays is located near the left side to provide selection of either low- or high-speed signals. In the low-speed mode, ATE signals directly provide the DUT inputs, permitting DC tests or functional testing up to 1 Gbps. In the high-speed mode, the multiplexed signals connect to the DUT inputs. Multipin low-speed connectors carry data signals from the ATE to the multiplexer inputs.

Figure 8 illustrates the Driver10-HT module's performance, showing data eye diagrams at 2.5 Gbps, as well as the minimum and maximum amplitudes. Both amplitude and DC bias are adjustable in this module, under the control of two analog voltage signals provided by the ATE. The range for both offset and amplitude is about 1 V and is limited by the output buffer used for the module. Rise and fall times are about 120 ps (20% to 80%). Faster edge rates are possible with silicon germanium (SiGe) buffers (see Figure 9, for example).

Figure 9a shows the Driver6 design's performance in a data eye diagram at 2.5 Gbps.[2] We took this measurement at the DUT input (after the signal has passed through relays and connectors). The measured jitter is 42 ps peak to peak and includes the oscilloscope trigger jitter (conservatively estimated at 20 to 30 ps). This leaves an eye opening of at least 0.87 unit interval (UI) at 2.5 Gbps. We measured the signal's rise time separately at about 60 ps (20% to 80%), including the bandwidth limitations of an 8-GHz resistive probe (for the measurements shown, we used a 5-GHz differential field-effect transistor probe). This is consistent with the expected SiGe output transitions of about 40 ps and some (small) rise time degradation through the relays and connectors.

Figure 9b shows the data eye when we injected an additional 100 ps of jitter, using the circuit shown in Figure 6b. Figure 9c shows the signal with about 200 ps of added

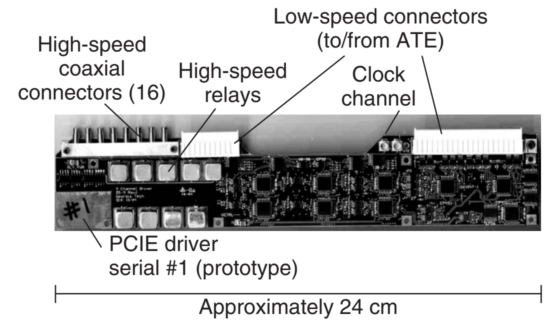
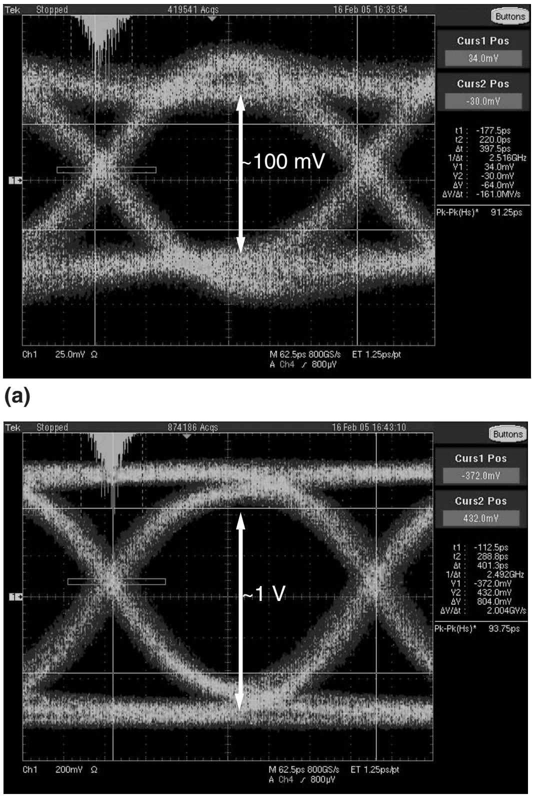

Low-speed connectors (to/from ATE)
High-speed coaxial connectors (16)
High-speed relays
Clock channel
#1
PCIE driver serial #1 (prototype)
Approximately 24 cm

Figure 7. Nine-channel (differential) Driver10-HT test module. The ninth channel provides the clock signal required by HyperTransport applications.



(a)

419541 Acqs 16 Feb 05 16:35:54

Curs1 Pos 34.0mV

Curs2 Pos -30.0mV

```
t1 :   -177.5ps
t2 :    220.0ps
Δt :    397.5ps
1/Δt :    2.516GHz
V1 :     34.0mV
V2 :    -30.0mV
ΔV :    -64.0mV
ΔV/Δt : -161.0MV/s
Pk-Pk(Hs)*  91.25ps
```

Ch1 25.0mV Ω M 62.5ps 800GS/s ET 1.25ps/pt A Ch4 ⟋ 800µV

(b)

874186 Acqs 16 Feb 05 16:43:10

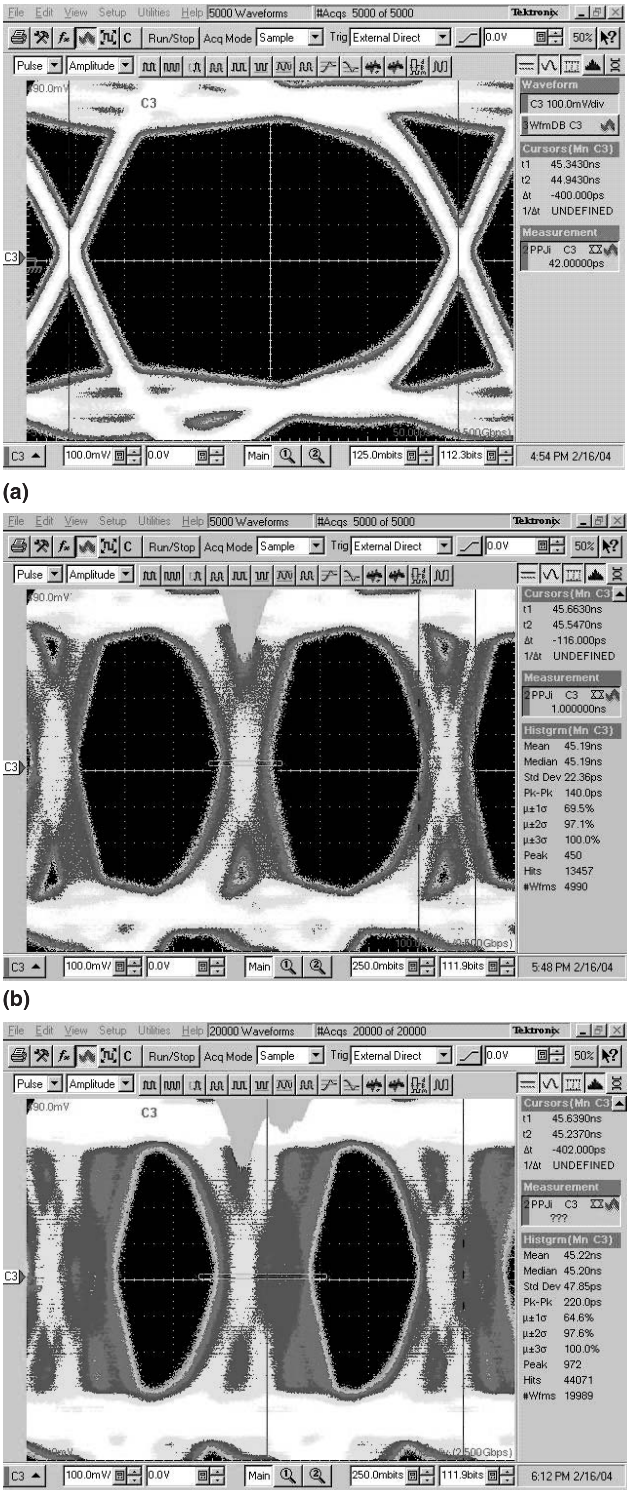
Curs1 Pos -372.0mV

Curs2 Pos 432.0mV

```
t1 :   -112.5ps
t2 :    288.8ps
Δt :    401.3ps
1/Δt :    2.492GHz
V1 :   -372.0mV
V2 :    432.0mV
ΔV :    804.0mV
ΔV/Δt :   2.004GV/s
Pk-Pk(Hs)*  93.75ps
```

Ch1 200mV Ω M 62.5ps 800GS/s ET 1.25ps/pt A Ch4 ⟋ 800µV

Figure 8. Example data eye diagrams from Driver10-HT at 2.5 Gbps: minimum programmable amplitude of ~100 mV (a); maximum amplitude of ~1 V (b).



(a)

5000 Waveforms #Acqs 5000 of 5000

Acq Mode Sample Trig External Direct 0.0V 50%

Waveform: C3 100.0mV/div; 3 WfmDB C3

```
Cursors(Mn C3)
t1    45.3430ns
t2    44.9430ns
Δt    -400.000ps
1/Δt  UNDEFINED
Measurement
2 PPJi  C3
   42.00000ps
```

C3 100.0mV/ 0.0V Main 125.0mbits 112.3bits 4:54 PM 2/16/04

(b)

5000 Waveforms #Acqs 5000 of 5000

Acq Mode Sample Trig External Direct 0.0V 50%

```
Cursors(Mn C3)
t1    45.6630ns
t2    45.5470ns
Δt    -116.000ps
1/Δt  UNDEFINED
Measurement
2 PPJi  C3
   1.000000ns
Histgrm(Mn C3)
Mean     45.19ns
Median   45.19ns
Std Dev  22.36ps
Pk-Pk    140.0ps
µ±1σ     69.5%
µ±2σ     97.1%
µ±3σ     100.0%
Peak     450
Hits     13457
#Wfms    4990
```

C3 100.0mV/ 0.0V Main 250.0mbits 111.9bits 5:48 PM 2/16/04

(c)

20000 Waveforms #Acqs 20000 of 20000

Acq Mode Sample Trig External Direct 0.0V 50%

```
Cursors(Mn C3)
t1    45.6390ns
t2    45.2370ns
Δt    -402.000ps
1/Δt  UNDEFINED
Measurement
2 PPJi  C3
   ???
Histgrm(Mn C3)
Mean     45.22ns
Median   45.20ns
Std Dev  47.85ps
Pk-Pk    220.0ps
µ±1σ     64.6%
µ±2σ     97.6%
µ±3σ     100.0%
Peak     972
Hits     44071
#Wfms    19989
```

C3 100.0mV/ 0.0V Main 250.0mbits 111.9bits 6:12 PM 2/16/04

Figure 9. SiGe Driver6 2.5-Gbps data eye diagrams with minimal (42-ps) total jitter injection (a), moderate (140-ps) total jitter injection (b), and large (220-ps) total jitter injection (c).

Table 2. Example receiver modules.

Design	Function	No. of channels	Fmax (Gbps)
Receiver1	1:4 demux	8 differential	2.0
Receiver2	1:4 sampler	12 differential	2.0
Receiver3	1:4 ReClk sampler	12 differential	2.5
Receiver4	1:2 sampler	24 single-ended	1.8
Receiver5	1:8 demux	4 differential	3.6
Receiver6	1:16 demux	2 differential	5.0
Receiver9-PCIE	1:4 demux	8 differential	2.5
Receiver10-HT	1:2 demux	9 differential	2.0

jitter. To obtain these results, we use the jitter injection circuit to dynamically shift the phase of a low-jitter ReClk signal, under the control of digital data provided by the ATE. The delay circuit in Figure 6b has a 10-ps resolution (per digital bit). The digital delay encoding is nominally in binary multiples of the 10-ps resolution. As the test sequence progresses and the ATE sequentially provides parallel data to the multiplexer, the ATE also sequences through a prescribed pattern of digital delay values (binary codes). The delay chip (in Figure 6b) then converts these codes to analog delay values in real time.

Because we can program any desired code word sequence, we can synthesize the generation of a wide variety of jitter effects. We often use a pseudorandom pattern, carefully constructed to match the Gaussian probability density function with a desired standard deviation. Bimodal, multimodal, and sinusoidal distributions are also used. For example, Figure 9b shows a normal Gaussian distribution, whereas Figure 9c is clearly bimodal.

Receiver modules

To support a wide variety of testing applications, we have developed several receiver modules, each implementing either demultiplexing or sampling methods. Table 2 lists examples. Receivers 1 through 6 support synchronous testing. They are phase-locked to a low-jitter (~1.3-ps rms) clock source, which is locked to the ATE system PLL. In synchronous applications, we also use the external source to clock the driver modules. The performance of these synchronous receiver modules has been described elsewhere[1,2] and generally matches the driver modules' performance (Fmax, timing accuracy, jitter, and so on). Because these designs use similar components to those of the driver circuits, we typically obtain comparable performance in the receivers. Designing the receivers is a little easier because we can generally obtain highly sensitive differential buffers more readily than high-performance output drivers. On the other hand, we must carefully control both the clock and data jitter.

The Receiver9-PCIE and Receiver10-HT designs specifically target our example application. Figure 10 shows the basic demultiplexing logic implemented for each receiver channel in PCI Express and HyperTransport applications. On the left side, differential data from the DUT goes through coaxial relays to a differential buffer. This buffer's outputs pass to demultiplexing or sampling logic circuits. The resulting data channels connect to the ATE, and provide the sampled data at slower rates suitable for real-time comparison in the ATE. Like the driver modules, the receivers include relays that permit signal switching directly into tester channels used for DC parametric measurements of DUT characteristics, such as output voltage sensitivity (VOL/VOH).

In the PCI Express bus, we encode the source-synchronous clock into each data signal, and phase-lock the receiver clock to logic transitions in the data itself. In the receiver modules, we accomplish this by splitting the received data between the demultiplexer and a clock data recovery (CDR) chip. The CDR chip is a commercially available part tuned to the 2.488-GHz frequency used for PCI Express. An internal voltage-controlled oscillator produces the high-speed clock, and the PLL synchronizes this clock's phase to the expected midpoint of the valid data window. Because the data is 8b/10b encoded, it contains sufficient transitions to keep the clock recovery circuit locked (this circuit can tolerate more than 2,000 repeated bits before losing phase lock).

In normal (functional) applications, the CDR chip also reclocks the received data (using the internally generated 2.488-GHz signal), thereby regenerating the maximum-width data eye openings. However, for our testing applications, we don't want to lose the incoming data-timing information in this way. Instead, we must often measure these delays to determine how much timing margin is available. Therefore, we include an adjustable delay line to set the recovered clock's phase with respect to the data before input to the demultiplexer, as Figure 10a shows. During a characterization test, the ATE sweeps through a range of digital delay values (with about 10-ps resolution) as the test is repeated. Then we can determine the data eye widths by examining these results, either numerically or with a graphical shmoo plot.

Because we synchronize the sampling clock to the incoming data, this test approach is very tolerant to jitter present in the incoming data. The CDR chip we use passes low-frequency jitter components, so that the clock signal tends to track the jitter present in the data signal. Therefore, we obtain the wide data eye opening rather than losing the data because of jitter in the receiver.

Figure 10b shows the PCI Express recovered clock for an example set of DUT output data. In this case, we measured the recovered clock signal at the inputs to the 1:4 demultiplexer with time-interval-evaluation instrumentation, which is more sensitive than oscilloscope measurements. Typically, the clock signal exhibits about 3 ps of random jitter, resulting in about 42 ps total jitter at a 10^{-12} bit error rate. To some degree, this random jitter correlates with the data jitter (as described earlier), especially the low-frequency (less than 50-MHz) components. However, some of the effect is due to high-frequency noise generated in our receiver modules. Experimentally separating these effects has been challenging and continues to be a topic of investigation.

Unlike the PCI Express clock, the HyperTransport clock is carried by a separate source-synchronous differential channel (rather than being encoded in the data signals). Figure 10c shows how the HyperTransport receiver uses this signal to sample and demultiplex the data channels. This approach is almost identical to that of PCI Express, except that it doesn't include the CDR technique (because the DUT provides the clock signal separately).

At first, the HyperTransport receiver's clocking approach sounds simpler than the PCI Express approach since it does not require clock-recovery circuits. However, it has some drawbacks. First, it relies entirely on the DUT for the very critical reclocking signal. If this signal is not working properly, then we must substitute a lower-frequency clock signal from the ATE (we use other switching logic for this). Perhaps of even more concern is

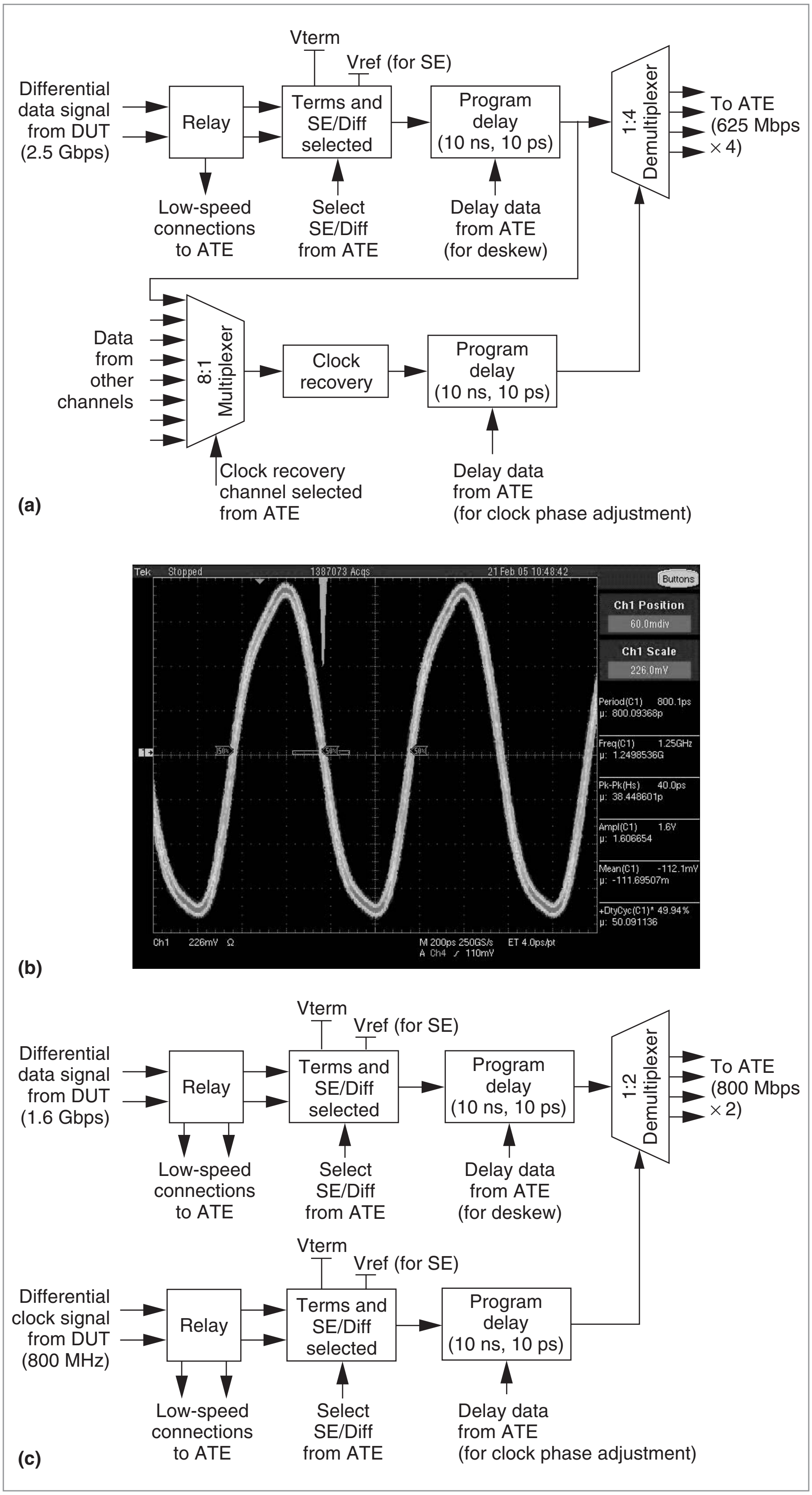

Vterm
Vref (for SE)
Differential data signal from DUT (2.5 Gbps)
Relay
Terms and SE/Diff selected
Program delay (10 ns, 10 ps)
1:4 Demultiplexer
To ATE (625 Mbps × 4)
Low-speed connections to ATE
Select SE/Diff from ATE
Delay data from ATE (for deskew)
Data from other channels
8:1 Multiplexer
Clock recovery
Program delay (10 ns, 10 ps)
Clock recovery channel selected from ATE
Delay data from ATE (for clock phase adjustment)
(a)
Tek Stopped 1387073 Acqs 21 Feb 05 10:48:42
Buttons
Ch1 Position 60.0mdiv
Ch1 Scale 226.0mV
Period(C1) 800.1ps μ: 800.09368p
Freq(C1) 1.25GHz μ: 1.2498536G
Pk-Pk(Hs) 40.0ps μ: 38.448601p
Ampl(C1) 1.6V μ: 1.606654
Mean(C1) -112.1mV μ: -111.69507m
+DtyCyc(C1)* 49.94% μ: 50.091136
Ch1 226mV Ω
M 200ps 250GS/s ET 4.0ps/pt
A Ch4 ⌿ 110mV
(b)
Vterm
Vref (for SE)
Differential data signal from DUT (1.6 Gbps)
Relay
Terms and SE/Diff selected
Program delay (10 ns, 10 ps)
1:2 Demultiplexer
To ATE (800 Mbps × 2)
Low-speed connections to ATE
Select SE/Diff from ATE
Delay data from ATE (for deskew)
Vterm
Vref (for SE)
Differential clock signal from DUT (800 MHz)
Relay
Terms and SE/Diff selected
Program delay (10 ns, 10 ps)
Low-speed connections to ATE
Select SE/Diff from ATE
Delay data from ATE (for clock phase adjustment)
(c)

Figure 10. Logic implemented by each receiver module channel for PCI Express application (a); recovered clock signal at 1.25 GHz, from 2.5-Gbps PCI Express data channel (b); logic implemented by each receiver module channel for HyperTransport application (c).

that the jitter present in the clock might not entirely correlate with jitter in the data signals. In the worst case (totally uncorrelated jitter), the effective random jitter can greatly diminish the eye width. The extent to which this becomes a problem depends in part on how well the DUT's HyperTransport output bus maintains correlated jitter across all parallel data channels and the clock. In the best case (perfectly correlated data and clock jitter), we can get wide eye openings.

Receiving nondeterministic data

So far in our testing applications, we have been fortunate that the DUT has incorporated testability features that let us force a deterministic output behavior. Therefore, we can predict when to sample the data at the receivers and exactly what data patterns to expect. However, some of the existing protocols allow device outputs that are nondeterministic in both timing and data pattern. For PCI Express, we account for the timing uncertainty by using a CDR to align clock and data. In general, however, there can still be uncertainty about which clock cycle represents the beginning of a sequential message. Most protocols (including those of PCI Express) handle this by using preambles and headers containing key symbols that signal the pending beginning of the payload data packets. In a normal application, the receiving logic includes state machines that recognize these sequences and use that information to align the clock to the payload data. Of course, this process is not trivial when the data streams are running at 2.5 Gbps or higher. So far, we have not provided that capability in our receivers. In some cases, loopback techniques that use the DUT receiver logic itself to capture its own output data can be applied. This is essentially a BIST approach. However, we fully expect that some of our receiver modules will need this function in the future. This is a feature currently under development and will likely be the subject of a future article.

Cost and maintainability for high-I/O applications

Because our solution is not yet commercially available, we cannot provide detailed cost comparisons. However, it is safe to say that the additional hardware costs are at least an order of magnitude lower than those of a high-end ATE platform and are somewhat lower than rack-and-stack solutions (which have their own limitations for high I/O situations). At the same time, our approach offers a more customized match to I/O testing needs. We have already applied our approach to DUTs with about 100 multi-GHz I/Os and were constrained mainly by the number of channels available in the host ATE, as well as the test head's mechanical requirements.

In principle, the approach is extendable to a far higher channel count, and we are developing modules that use FPGAs to replace much of the ATE functionality (thereby eliminating the channel count constraint). Redesigning the test head's mechanical arrangement to accommodate more channels is a nontrivial but very feasible engineering effort. The cost of maintaining the approach is admittedly substantial, but it is not excessive compared with the alternative (high-end ATE). The wide-ranging flexibility of our approach allows case-by-case optimization of economic and performance trade-offs. Compared with heavy investment in "super ATE," our approach requires less up-front capital investment, while requiring dedicated design and fabrication effort for each DUT.

A SECOND PHASE of this project will address the nondeterministic nature of PCI Express. Our plan is to use a match-loop synchronization scheme that repeats a training pattern to detect the beginning of a packet and then captures payload packets and postprocesses the results. Postprocessing will detect and extract skip order sets so that we can reconstruct the DUT data stream before pattern-signature comparison. An alternative is to detect the start of the pattern in real time, which is very challenging at these high data rates. Instead, we can demultiplex the serial data to slower (but wider) parallel words and make comparisons in a pipelined and parallel fashion. Although these techniques are feasible if integrated on a custom monolithic chip, they are challenging to implement using off-the-shelf components. However, the advent of high-performance FPGAs is helping us to address this challenge. ■

Acknowledgments

We conducted the work described in this article as a joint R&D project between Georgia Tech and IBM Canada. We are grateful to IBM engineers for their efforts in designing, constructing, and testing the thermal management systems. We conducted the system-level experiments using equipment at the IBM Canada facility.

■ References

1. D.C. Keezer, D. Minier, and M.C. Caron, "A Production-Oriented Multiplexing System for Testing above 2.5 Gbps," *Proc. Int'l Test Conf.* (ITC 03), IEEE Press, 2003, pp. 191-200.

2. D.C. Keezer et al., “Modular Extension of ATE to 5 Gbps,” *Proc. Int’l Test Conf.* (ITC 04), IEEE Press, 2004, pp.748-757.
3. D.C. Keezer, “Multiplexing Test System Channels for Data Rates above 1 Gb/s,” *Proc. Int’l Test Conf.* (ITC 90), IEEE Press, 1990, pp. 362-368.
4. L. Ackner and M.R. Barber, “Frequency Enhancement of Digital VLSI Test Systems,” *Proc. Int’l Test Conf.* (ITC 90), IEEE Press, 1990, pp. 444-451.
5. D.C. Keezer et al., “Terabit per Second Automated Digital Testing,” *Proc. Int’l Test Conf.* (ITC 01), IEEE Press, 2001, pp. 1143-1151.
6. D. Wimmers, K. Sakaitani, and B. West, “500 MHz Testing on a 100 MHz Tester,” *Proc. Int’l Test Conf.* (ITC 94), IEEE Press, 1994, pp. 273-278.
7. M. Shimanouchi, “Periodic Jitter Injection with Direct Time Synthesis by SPP ATE for SerDes Jitter Tolerance Test in Production,” *Proc. Int’l Test Conf.* (ITC 03), IEEE Press, 2003, pp. 48-57.
8. T. Yamaguchi et al., “Effects of Deterministic Jitter in a Cable on Jitter Tolerance Measurements,” *Proc. Int’l Test Conf.* (ITC 03), IEEE Press, 2003, pp. 58-66.
9. K. Taylor et al., “CMOS Built-in Test Architecture for High-Speed Jitter Measurement,” *Proc. Int’l Test Conf.* (ITC 03), IEEE Press, 2003, pp. 67-76.
10. J. Hops et al., “Non-Deterministic DUT Behavior during Functional Testing of High Speed Serial Busses: Challenges and Solutions,” *Teradyne Test Times Newsletter*, 2004; http://www.teradyne.com/std/newsletter/issue04/NDD_solution_for_hss-rev4.4.pdf.

David C. Keezer is a professor of electrical and computer engineering at the Georgia Institute of Technology. His research interests include the design and test of high-performance electronic systems. Keezer has a BS in physics from the University of California, Berkeley; an MS in applied physics from the California Institute of Technology; an MBA from the Florida Institute of Technology; and a PhD in electrical engineering from Carnegie Mellon University. He is a senior member of the IEEE.

Dany Minier is a test engineer at IBM Canada. His research interests include high-speed test solutions with ATE. Minier has a BEng in electrical engineering from the University of Quebec.

Patrice Ducharme is a test engineer at IBM Canada. His research interests include high-speed test solutions, characterization with ATE, thermal management, and high-power delivery. Ducharme has a BEng in electrical engineering from Ecole de Technologie Superieure (Montreal, Quebec) and an MS in project management from Sherbrooke University.

■ Direct questions and comments about this article to David C. Keezer, Georgia Institute of Technology, School of Electrical and Computer Engineering, Atlanta, GA 30332-0250; dkeezer@ece.gatech.edu.

For further information on this or any other computing topic, visit our Digital Library at http://www.computer.org/publications/dlib.