# Itanium™ Processor Clock Design

Utpal Desai[1], Simon Tam, Robert Kim, Ji Zhang, Stefan Rusu

Intel Corporation, M/S SC12-502, 2200 Mission College Blvd, Santa Clara, CA 95052

## ABSTRACT

The Itanium processor is Intel's first 64-bit microprocessor [1] and features a highly parallel architecture fabricated using the 0.18um process. This higher integration of features requires a significant silicon real estate and high clock loading. These factors, coupled with more prominent on-die variations because of reduced device geometries, call for special techniques to manage the clock design. The Itanium processor employs very well balanced clock routing along with distributed deskew buffers (DSK) to achieve low skew. The Itanium™ processor also includes additional features to aid performance tuning and timing debug. This paper highlights the salient features of the Itanium processor clock design and presents clock characterization data from initial silicon.

## KEYWORDS

Itanium™ processor, IA-64, clock distribution, deskew, On-Die-Clock-Shrink.

## 1. INTRODUCTION

The Itanium processor is Intel's first 64-bit microprocessor and features a highly parallel architecture fabricated using the 0.18um process. This high level of integration requires significant silicon real estate and results in high clock loading. The larger die size and the smaller feature sizes because of the 0.18u process result in higher on-die variations. The combination of the above two factors causes the skew due to on-die variations to be a bigger component of the total clock skew on the die. Many techniques have been developed and utilized in prior processors to manage clock skew. These include the use of PLLs or DLLs, passively tuned clock trace lengths and matching end loads, use of balanced routing structures like binary trees and H-trees [2] and fully shielded clock traces to reduce inductive effects and lateral cross-coupling noise. For the lower performance designs, it was sufficient to use passively tuned clock networks in conjunction with the PLL to achieve performance targets. However, the passively tuned clock networks cannot correct for skew that results because of on-die variations. For this reason, the Itanium™ processor implements active deskewing of the clock on top of the passively tuned network.

The Pentium®-III processors first added the capability to reliably manipulate the clock edges in-situ, which resulted in dramatic improvements in the efficiency of timing debug for that processor. Therefore, the Itanium™ processor's clock design also incorporates a similar On-die-clock-shrink feature to achieve this capability.

## 2. CLOCK TOPOLOGY

Figure 1 shows the high level Itanium™ processor clock generation and distribution scheme. The core PLL receives differential clock inputs running at the bus clock frequency and generates a high frequency clock at twice the core clock frequency [3]. A divide-by-two circuit generates the high frequency core clock and the reference clock using the 2X-frequency clock from the PLL. Both these clocks, the core clock and the reference clock, are routed from the PLL via a balanced H-tree to 8 deskew clusters, each of which contains four distinct deskew buffers (DSK). The deskew buffer is a digitally controlled analog delay line, whose function is to detect and eliminate the skew between any two clocks and will be described in details later. The reference clock route stops at the deskew clusters. The core clock, on the other hand is routed as an input to the DSKs within a deskew cluster and generates output clocks shown as gclk in Figure 1. The total die area is partitioned into 30 regions, and therefore only 30 out of the 32 gclk signals are used to generate the clocks required by a clock region. The gclk output from a DSK is buffered by bank of buffers called the regional clock drivers (RCD) located at the top and bottom of a clock region and distributed over the clock region via a clock grid. The circuits within a clock region tap directly into the overlaying grid to generate the local clocks

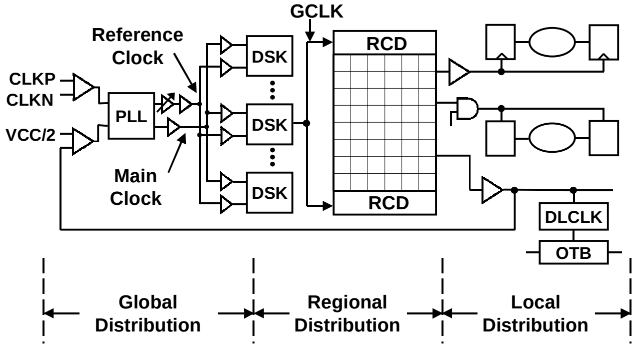

**Figure 1: Clock Distribution Topology for the Itanium™ Processor**

---

[1] E-mail address: utpal.desai@intel.com

*ISPD 2000*, San Diego, CA.
ACM 1-58113-191-7/00/0004.

required within the region. In summary, the clock distribution can be broken up into three sections:

1. The Global distribution comprising the clock distribution from the PLL to the DSKs,
2. The Regional distribution comprising the clock distribution from the DSKs to the regional clock grid, and,
3. The Local distribution comprising the local clock buffers consuming the clock from the regional grid to drive the clocked elements within the region.

The core PLL runs off a separate analog VCC, which is filtered by an off-chip LC filter. The rest of the clock distribution runs off the regular digital VCC. All the global clock routes, from the PLL output to the 30 clock grids, only use higher level metals, and are fully shielded. For an Itanium processor, the transition time of the clock signal is comparable to it's propagation delay; therefore, all clock routes are modeled using distributed LRC models [4], and optimized to minimize skew and mean delay. Narrower clock traces, shielded by power rails, are used to provide good return path and to minimize inductive effects.

## 3. THE DESKEW ARCHITECTURE

In the above architecture, the function of the DSKs is to eliminate any skew that exists between the 32 clocks at the clock grid. To achieve this, it is necessary to route a clock trace (feedback clock) from each of the regional grids back to the DSK where they can be compared against a common reference to determine the skew. The reference clock is the common reference that is used for this purpose. A phase detector inside the DSK detects the skew between the rising edges of the reference clock and the feedback clock and issues a correction to eliminate any skew that might exist between these two clocks. This is achieved by varying the delay of the core clock through the DSK, either by pulling in or pushing out the clock at the clock grid. After a successful deskewing operation, the only residual skew that remains is the sum of the skew between the 30 reference clocks, the skew between the 30 feedback clock traces, the uncertainty of the phase detector itself, and the discrete delay step size of the DSK. The clock network is passively tuned to eliminate any DC skew between the reference clocks and between the feedback clock traces. The addition of the dynamic DSK architecture also eliminates skew that occurs due to systematic process variations in the clock distribution network, a benefit not achievable by passive tuning of the network.

## 4. DESKEW BUFFER DESIGN

The block diagram of a DSK is shown in Figure 2. The design consists of a local controller and a digitally controlled analog delay line represented by the Delay Circuit in Figure 2. The local controller detects the phase difference between the feedback clock and the reference clock and issues an adjustment to vary the delay through the analog delay line.

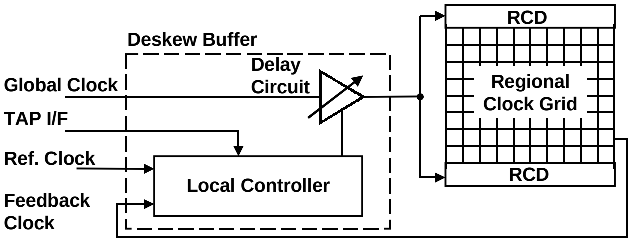

**Figure 2: Itanium™ Processor Deskew Buffer Architecture.**

Figure 3 shows the design of the digitally controlled analog delay line. The design consists of a four inverter buffer stage, a 20 bit Delay control register, with a Test Access Port (TAP) Interface and two banks of 20 passive loads which are connected to the intermediate nodes of the four inverter delay chain via a 20 bit digital switch. The first two stages of the inverter chain are CMOS inverters. The fourth stage, which is the output driver, is a modular design consisting of 12 parallel inverters which allows us to configure it's drive strength in 12 distinct step sizes. The third stage is used to program the drive strength of the final driver. The control bits in the 20 bit register control the digital switches, which determine the number of passive loads that get connected to the intermediate nodes of the delay chain. The 20-bit register deploys a thermometer code, where a 1 is shifted in from the left to speed up the delay line, and a 0 is shifted in from the right to slow down the delay line. The TAP interface allows us to read out the values from the 20-bit control register and also overwrite the default values with new settings if so desired.

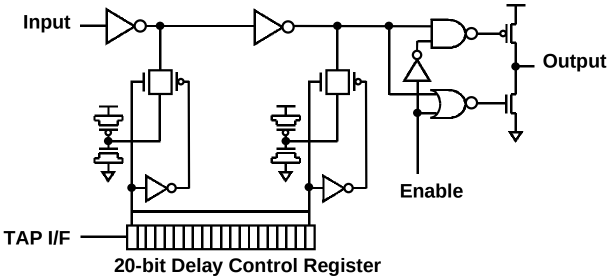

**Figure 3: Deskew Buffer Analog Delay Line**

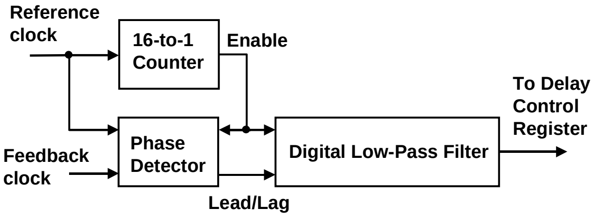

**Figure 4: DSK Local Controller Architecture**

Figure 4 shows the block diagram of the Local Controller located inside the DSK. It consists of a 16 bit counter, a phase detector and a low pass filter. The two inputs to the local controller are the reference clock and the feedback clock from the regional clock grid. The phase detector block detects the lead or lag relationship between these two clocks every 16 cycles. The digital low pass filter samples four consecutive lead/lag samples from the phase detector and issues a correction only if they are all identical. The correction is issued in the form of one bit, which pushes in a 0 or a 1 into the 20 bit delay control register. By sampling every 16 cycles, we allow the phase detector enough time to resolve any meta-stability, and issue a definitive lead/lag signal. The digital low-pass filter suppresses spurious errors that might result from clock jitter, preventing thrashing.

The delay across the analog delay line can be varied in 20 linear steps for a total of 170 ps. The TAP interface provides the ability to manually control the delay across the DSK. This provides the additional capability to intentionally skew the clocks between any two clock regions to identify timing paths (min and max) during silicon debug.

## 5. REGIONAL AND LOCAL CLOCK DISTRIBUTION

The clock grid over a clock region is driven by a set of buffers (RCDs) located at the top and bottom of a clock region. The Itanium™ processor clock grid is designed in metal 4 (M4, X direction) and metal 5 (M5, Y direction). The metal grid templates for these two layers are designed to include reserved clock tracks up front to ensure that the desired metal tracks are always available. The clock grid utilizes upto 3.5% of available M5 and upto 4.1% of available M4 routing over a region. Only the M5 wires are driven by the RCDs, the M4 wires are only for the purpose of equalization and interfacing with the M3 clock lines for local consumption of the clock within a region. Locally, every block taps off the M4 grid wire and buffers the clock once prior to driving the latches and flip-flops. Special types of clocks for more advanced circuit design styles like OTB domino are generated locally as well by using clock-chopping buffers. Since only the M5 grid wire is driven by the RCDs, to control skew within a grid, the clock regions are restricted in the Y dimension (M5 direction) but are allowed to be different in the X-dimension (M4 direction).

The design of the RCDs is modular. A series of RCD buffer templates are created to fit a multiple of the M5 grid template pitch. The delay though the various RCD templates are passively equalized by tuning the interconnect wire widths and the strength of the intermediate drivers. The placement pitch of the clock drivers within a RCD template is the same as that of the reserved clock tracks in the M5 grid. This enables the outputs of the RCD drivers to be ported up directly to connect to the M5 wires of the clock grid without having to waste lower level metals in the X-direction. This design also provides predictable free space for the lower level metals to pass through the RCDs in the Y direction. Sufficient decoupling capacitors are included in the RCD templates to minimize power supply noise. Depending on the X-dimension of a clock region, multiple templates of the RCDs are assembled side by side to create a bank of buffers that could be dropped in place to interface with the M5 wires constituting the grid. The input clock to the RCDs from the DSK is routed using binary distribution and is passively equalized between all clock regions.

## 6. REFERENCE CLOCK DESIGN

The feedback clock is compared against the common reference clock inside a DSK to detect the skew that it might have with respect to the reference clock. Therefore, it is critical that the skew between the reference clocks at each of 30 DSK phase detector inputs is tuned out, since any skew between these reference clocks becomes a residual skew that cannot be overcome by the DSK. The positioning of the reference clock edge is very critical in the above architecture. If the reference clock is too fast or too slow compared to the feedback clock, the DSK will not have sufficient range to compensate for the skew difference between the reference clock and the feedback clock.

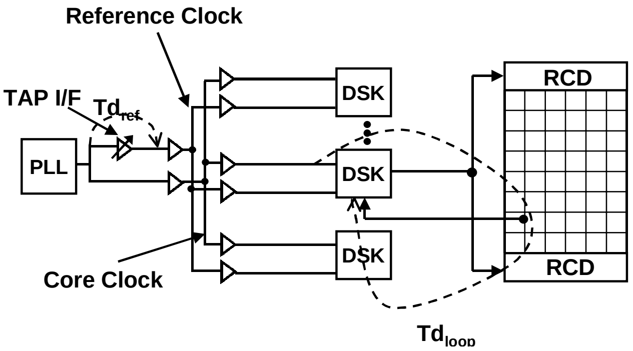

**Figure 5: Itanium™ Processor Clock Distribution Network**

As shown in Figure 5, the reference clock is derived from the core clock inside the PLL by inserting a delay line in series, after tapping off the core clock. This reference clock signal is then buffered and routed as a balanced H-tree, identical to the core clock, to the DSK phase detector inputs. The magnitude of the reference clock delay line, $Td_{ref}$, is equal to the average of the loop delays, measured from the DSK input, through the RCD and back to the phase detector input inside the DSK. $Td_{loop}$ in Figure 5 represents this delay. The delay of the reference clock programmable line was optimized such that under a fully deskewed condition, each of the 30 DSKs would operate at their mid point delay setting (10 out of 20 bits are on). The TAP interface also allows manual adjustments to be made to $Td_{ref}$.

The overall skew of the system is determined by the following four factors:

1. The skew between the reference clocks at the 30 DSK phase detector inputs.
2. The skew between the wire delays of the 30 feedback clocks at the DSKs phase detector inputs.
3. The skew across the regional clock grid.
4. The discrete step size of the DSK.

Since the reference clock routes stop at the DSKs, it's span is about half that of the core clock. The end loads seen by the reference clock inside the DSKs are equalized as well. The feedback clock path length from the clock region back to the

DSK is a short point to point route and is tuned between all 30 clock regions.

Figure 6 shows the measured clock skew on silicon, measured on the clock grid. These measurements were made using a laser voltage prober (LVP). [5]

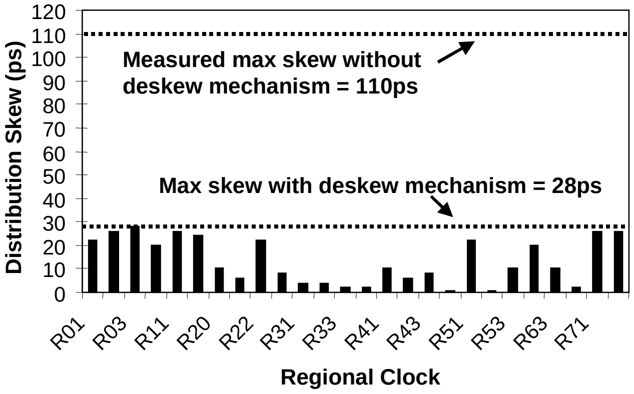

**Figure 6: Experimental skew measurements**

The measured worst case clock skew between any two clock regions is 28ps. If deskewing is turned off, the measured worst-case clock skew between all clock regions is 110 ps.

## 7. CLOCK SKEW AND TIMING ANAYSIS

The clock topology adopted for the Itanium™ processor leads to four distinct categories for skew values used for the timing analysis flows. Figure 7 identifies the four categories, depending on the relative location of the driver and the receiver. The smallest skew is for driver and receiver clocked elements being driven off the same local clock buffer (LCB) within a clock region (common LCB). The second best scenario for skew is when the driver and receiver are driven by different LCBs located within the same clock region (common RCD). The skew value for this category is worse than the first category by an amount equal to the skew across the clock grid itself.

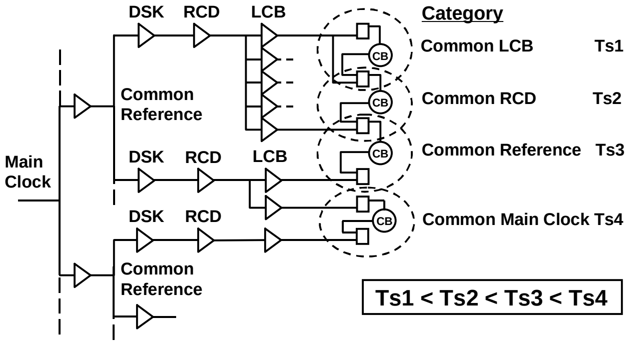

**Figure 7: Clock Skew Timing Budget.**

The skew between the reference clock between the four DSKs within a deskew cluster is smaller than the skew between the reference clocks between any two deskew clusters, because of their proximity on the die, and common distribution up to the input of the DSK cluster. Therefore, the third category for skew is defined when the driver and receiver are located in different clock regions, where the two regions are served by the same deskew cluster (Common Reference). The worst skew results when the driver and receiver belong to different clock regions, where the two clock regions are served by different deskew clusters (Common Main Clock).

## 8. ITANIUM™ PROCESSOR ON-DIE-CLOCK-SHRINK ARCHTECTURE

With the advent of C4 technology, the power variations across the die reduced significantly. However, with flip-chip design, accessibility to probing signals on the die has become a challenge. Therefore, the ability to manipulate clock edges in-situ so as to alter its instantaneous frequency at a certain predetermined time is of significant value during silicon debug. This on-die-clock-shrink (ODCS) feature [6] has quickly become a widely accepted feature among all Intel processors.

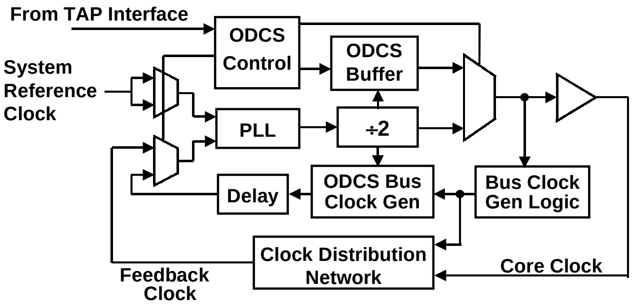

**Figure 8: Itanium™ Processor On-Die-Clock-Shrink Architecture**

Figure 8 shows the architecture of the Itanium™ processor's clock generation unit where the ODCS block is incorporated. The ODCS block consists of the ODCS buffer and the ODCS control logic. Its operation allows for the stretching and the shrinking of a clock cycle or its high or low phases at a predetermined time.

**Table 1 : ODCS Setting Example: Shortening high phase for the N+1 cycle (Figure 8)**

| State | Rise Setting (arbitrary time unit) | Fall Setting (arbitrary time unit) |
|---|---|---|
| Current | 10U | 10U |
| N+1 | 10U | 8U |
| N+2 | 8U | 8U |
| N+3 | 8U | 8U |

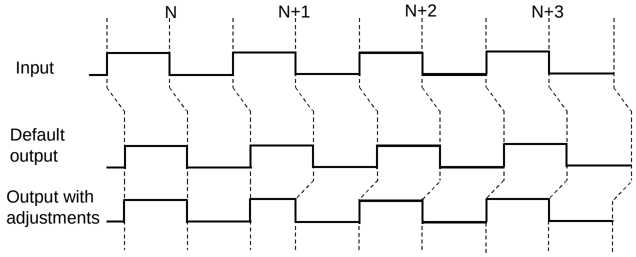

**Figure 9: Manipulated Waveform with a shortened High Phase at the N+1 Cycle**

An example of the operation is shown in Figure 9 based on the settings shown in Table 1. In this example, the ODCS delay buffer output has 10 U (arbitrary time unit) of rise delay and 10 U of fall delay from the buffer input (as seen at the Current Rise and Current Fall settings). The rise and fall settings for the N+1cycle are 10U and 8 U respectively. At the N+2 cycle, the rise and fall settings remained at 8U respectively. The resulting waveform is illustrated in Figure 6. It can be seen that the clock's high phase at the N+1 cycle has been reduced by 2 U. At the subsequent times the settings are at the default value, which means that the original duty cycle is restored.

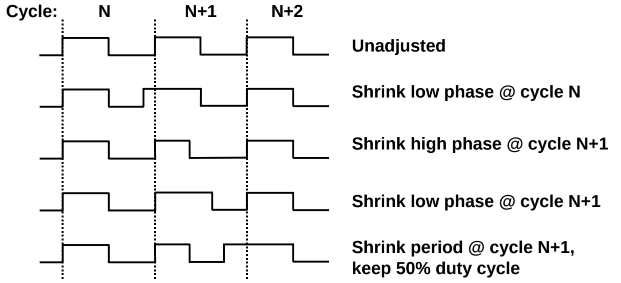
Cycle:
N
N+1
N+2
Unadjusted
Shrink low phase @ cycle N
Shrink high phase @ cycle N+1
Shrink low phase @ cycle N+1
Shrink period @ cycle N+1, keep 50% duty cycle

**Figure 10: Some Clock Waveforms that can be created using the ODCS feature.**

Figure 10 demonstrates several waveforms that can be created using ODCS. Silicon measurements have shown that the total range for manipulating an edge during ODCS mode is 200 ps. Fourteen discrete and linear step sizes are available over the total range of 200 ps, with each step size being about 16ps.

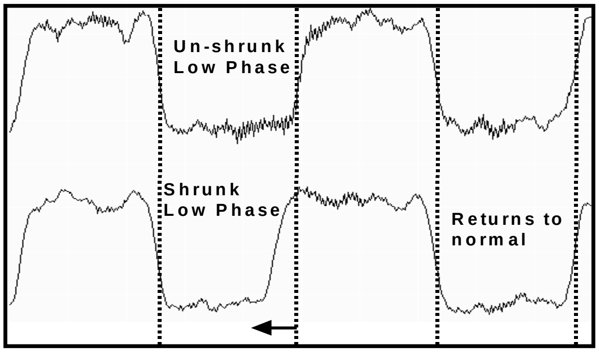

**Figure 11: Shrinking a low phase using ODCS**

Figures 11 and 12 show manipulated clock waveforms obtained from the Itanium™ processor silicon, demonstrating a shrunk low phase and a shrunk high phase.

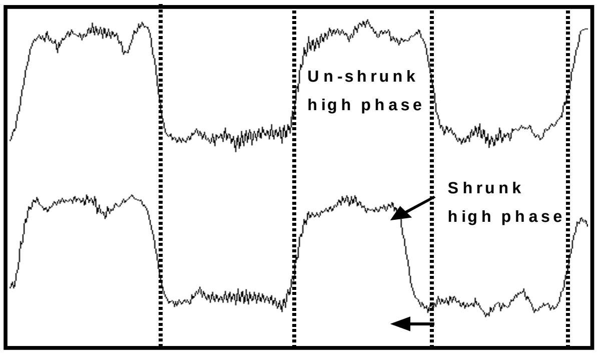

**Figure 12: Shrinking a high phase using ODCS**

## 9. SUMMARY

In summary, the Itanium™ processor utilizes an active deskewing scheme to achieve a low clock skew. This scheme compensates for process variations which is not possible using a passive scheme. Total measured skew between the 30 clock regions in the Itanium processor is 28 ps. In contrast, the measured skew with the deskew buffer architecture turned off, would have been 110 ps. The local clock architecture also supports a high frequency design, where, intentional clock skewing and time borrowing are used to enhance speed. The ODCS architecture has been used successfully by the silicon debug team to uncover timing paths to within 16ps. The ability to intentionally skew the clocks between two clock regions via the TAP interface to the DSKs has allowed the design team to tune top inter-clock-region timing paths.

## 10. REFERENCES

[1] Singer, G., Rusu, S., “The First IA-64 Microprocessor: A Design For Highly-Parallel Execution”, ISSCC Digest of Technical Papers, 2000.

[2] Restle, P.J. et. al, “Measurement and Modeling of On-Chip Transmission Line Effects in a 400 MHz Microprocessor”, IEEE Journal of solid-state circuits, vol. 33, no. 4, April 1998.

[3] Young, I. et. al., “A 0.35um CMOS 3-880MHz PLL N/2 Clock Multiplier and Distribution Network with Low Jitter for Microprocessors”, ISSCC Digest of Technical Papers, 1997.

[4] Deutsch, A. et. Al, “When are transmission-line effects important for on-chip interconnections?”, IEEE Transactions on Microwave Theory and Techniques, vol. 45, no 10, pt. 2, Oct. 1997.

[5] Eiles, T. et. al, “Optical Probing of Flip-Chip-Packaged Microprocessors”, ISSCC Digest of Technical Papers, 2000.

[6] Fischer, S. et. al, “A 600 MHz IA-32 Microprocessor with Enhanced Data Streaming for Graphics and Video”, ISSCC Digest of Technical Papers, 1999.