Technical Visualizations in VLSI Design

Phillip J. Restle
IBM T. J. Watson Research Center,
P. O. Box 218
Yorktown Heights, NY 10598, USA
914-945-2697
restle@us.ibm.com

ABSTRACT
Visualization techniques were applied to several different types of VLSI design and simulation data. A number of different visualizations have been tried, with varying results. Examples include 3D visualization of voltage and currents from fullwave interconnect analysis, on-chip clock distribution networks, chip/package power supply noise analysis, wire congestion, chip layout imaging, and static circuit tuning. The goals, successes, and failures of these examples will be discussed, along with some unexpected benefits from our ability to easily see patterns in complex visualizations.

General Terms
Design, Human Factors.

Keywords
Visualization, VLSI Design, Interconnect, Inductance, Clock Distribution, Circuit Optimization.

1. INTRODUCTION
As VLSI design becomes more sophisticated and complex it becomes increasingly important to analyze and process huge amounts of data into understandable results that can be clearly communicated. Largely because of previous design successes, every VLSI designer and tool developer now has access to workstations with formidable power to quickly create sophisticated visualizations of their data. In spite of this, designers still rely primarily on traditional 2-D visualization methods (with incremental improvements) for most work. Only occasionally have more sophisticated images or animations been created in the field of VLSI design, and they were created by visualization specialists for public relations or advertising. In this paper, several examples will be shown of experimental technical visualizations created primarily by VLSI designers or tool developers to help understand a problem or convey a concept (not to sell a product). These attempts produced everything ranging from surprising results, valuable insights, and obvious enjoyment, to thinly veiled contempt from respected colleagues.

DAC 01, Las Vegas, Nevada

The reality is that most engineers require no visualizations beyond traditional line drawings, waveform plots, and the use of a 2D layout editor to complete extremely complicated design challenges. On the other hand, while more sophisticated visualizations are rarely necessary, they can occasionally be valuable beyond the original motivations. Independent of any technical merit, some people love to see, explore, and create new visualizations, while others have no interest in them at all.

2. Examples

2.1 Interconnect Voltage and Current
The first five examples all use variations of a new visualization method developed to help display all the currents and voltages in simulations of complex interconnect structures. These 3D animations were initially designed to aid the author to develop physical intuition about on-chip transmission line issues such as inductive return-path design, inductive coupling, and power grid integrity, after measurements proved these effects important [1,2]. In fact the visualizations were quite helpful, and proved an effective and enjoyable way for chip designers to learn the basics of return-path design for high frequency on-chip interconnects. The extraction and simulation were performed with an innovative fullwave 3D PEEC (Partial Element Equivalent Circuit) simulator [3,4].

The visualization method used starts with a simple 2D layout view of wires in the X-Y plane. This layout view is then distorted to show the voltages and currents everywhere at a single point in time. Figure 1 shows such a visualization of a structure containing a 1 cm long copper on-chip signal wire surrounded by a power grid.

In this case the Y-axis is exaggerated since this layout is actually 70 times as long as it is wide. The Z-axis is used to represent voltage: every point on each wire is displaced in the Z-direction a distance proportional to the voltage at that point on the wire. Next, the diameters of the wires are distorted so that the diameter of each wire is proportional to the current at each point in the wire. In figure 1, most wires shown are part of the GND grid, and have a Z coordinate near 0 Volts. Two fairly straight wires near the edges of the structure are VDD wires, and thus have a Z coordinate of approximately 1.8 Volts. A single signal wire is being switched from GND to VDD at this snapshot in time. The near end has already been pulled up most of the way to 1.8 V, but the far end is still at GND. Color is used to show the sign of the current flow relative to the X, Y coordinate axes (see color figures on CDROM, and for animations see [13]).

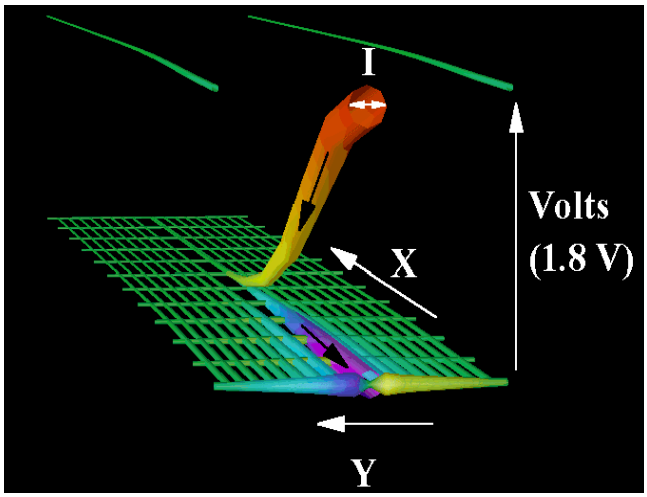

Figure 1. A 3-D visualization of currents and voltages. The Z-axis is voltage, and the wire diameter and color represent current.

2.2 Discontinuous Power Grid

In this example (figure 2) half of the GND wires nearest the signal wire were cut, representing a partial discontinuity in the power grid. Several features are clearly observable in the resulting visualization (especially when animated) that are not obvious from more traditional selected-waveform plots. We clearly see how the return current detours around the cut in the power grid, and a disturbance (300 mV) seen on the cut power wires due to both capacitive and inductive coupling to these GND wires.

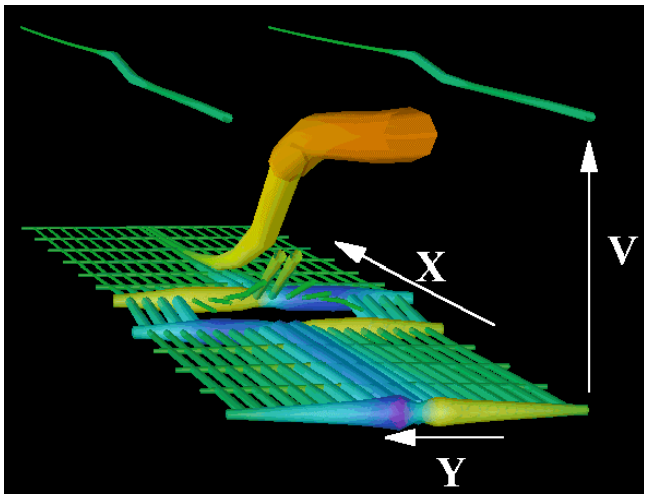

Figure 2. Current pattern due to power-grid discontinuity show the complexity of determining

2.3 Debugging Interconnect Simulations

Creating layouts or netlists for 3D interconnect simulations is a complex and error prone task. Traditionally, only a few voltages and currents are studied from a simulation containing thousands of nodes. It is very easy to make mistakes that result in subtle inaccuracies. One unintentional but valuable result of rich visualizations is that many errors are often found easily when all the data is visualized. In figure 3, examples of three mistakes are shown, all caused by missing or extra vias: 1) a VDD-GND short, 2) a floating wire, and 3) some partially-connected GND wires. Although this example is a fictional, real errors of these three types were actually found using similar visualizations.

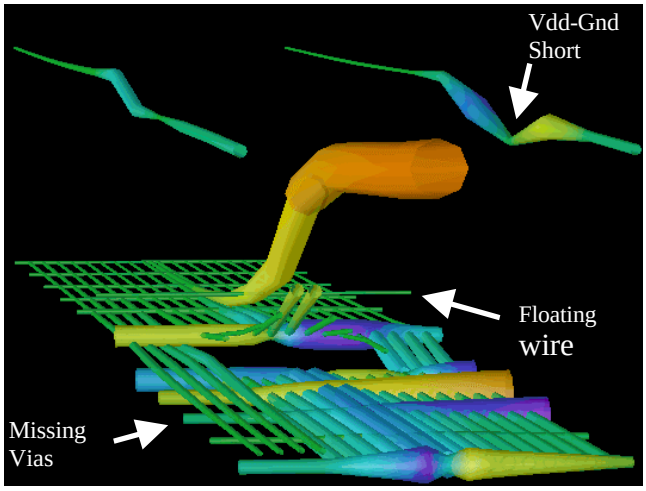

Figure 3 An example with three layout errors of types found using this visualization methods.

2.4 Simple Clock Mesh

The same visualization method was also useful for understanding and comparing high frequency clock distribution network strategies. Although many of these visualizations were originally prepared primarily for tutorial presentations, several of them resulted in unexpected insights.

One simple clock distribution network consists of a mesh of wire [5]. This mesh is then switched at the clock frequency using one or more lines of drivers. As clock frequencies increase, the interconnect delay between the drivers and the center of the mesh adds to the clock skew. Figure 4 shows an animation frame of such a network operating at 2 GHz, showing the center of the mesh lagging behind the edges. Resonance effects are easily observed and also degrade skew and signal integrity.

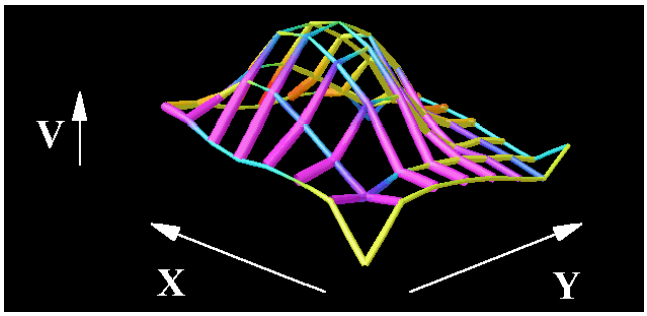

Figure 4. A 5 mm by 5 mm global clock mesh with drivers at all 4 edges shows significant skew when operated at 2 GHz.

2.5 Symmetrical H-trees

A more traditional clock distribution method employs some form of tree networks. When the clock load capacitances being driven are uniformly distributed, a symmetric H-tree as shown in figure 5 results in zero skew. Figure 6 shows a snapshot in time of the same tree driving more realistic non-symmetric loads, resulting in significant skew. In this case, coloring by current direction

was not useful since current directions are variable but relatively obvious, so the coloring was done using current magnitude only.

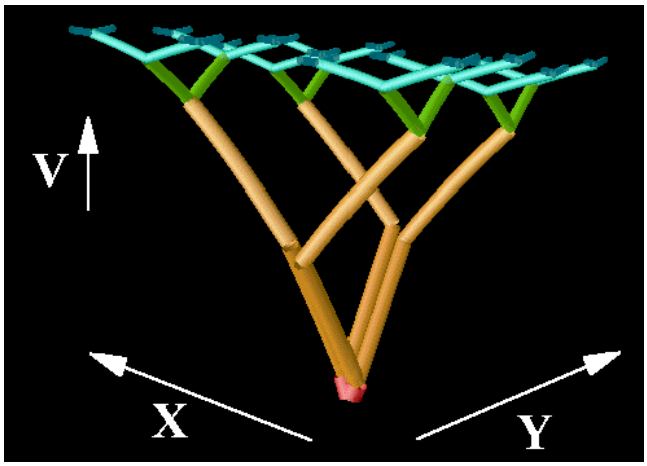

Figure 5. A snapshot in time of a symmetric tree driving a uniform load distribution.

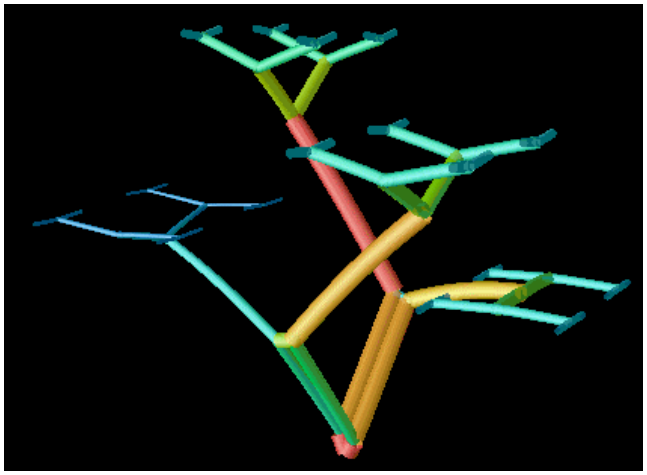

Figure 6. Tree driving non-uniform loads.

2.6 Tunable Trees

In all visualizations above, the Z-axis represented voltage, with different times shown on different frames of an animation. To focus on skew, it is more useful to create a single visualization (rather than an animation) using the Z-axis to represent delay (rather than voltage). The delays are defined using the VDD/2 crossing times. Figure 7 shows such a visualization made for a network similar to the case of figure 6, where widely varying loads produce different delays at different tree end-points. Figure 8 shows the same network after a wire-width tuning tool was used. One feature noticed from this visualization was the negative delay for one path in figure 8 (due to transmission-line effects).

An animation was made where subsequent frames of the animation represented tuning iterations of the wire-tuning tool (each frame representing a different network), rather than time steps in the operation of a single circuit.

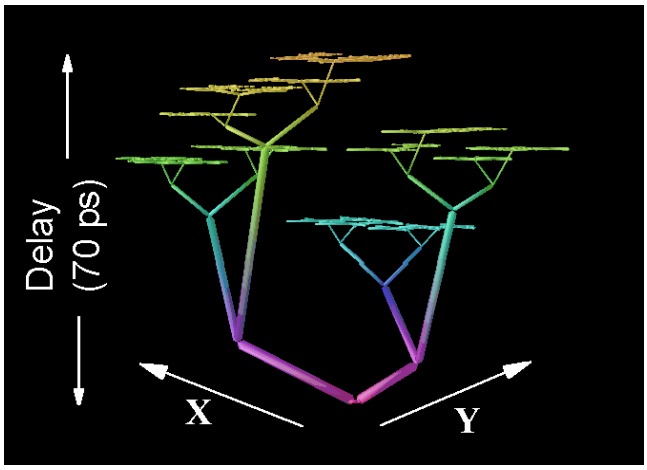

Figure 7. A tree driving a non-uniform load distribution where significant skew is observed between different branch end-points. Unlike previous figures, the Z axis represents delay rather than voltage, and the wire width shown represents the designed wire width, rather than current.

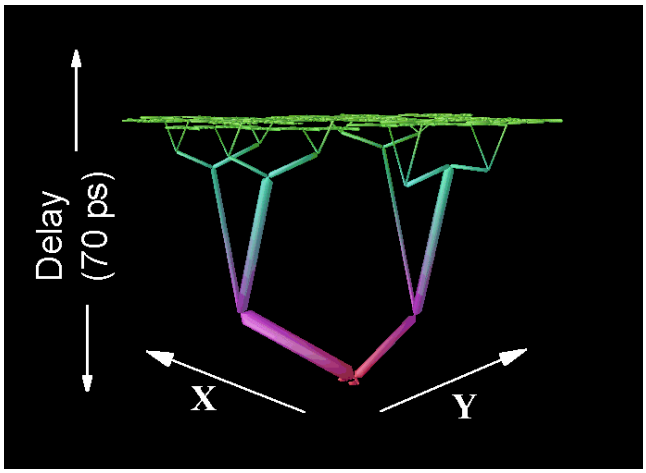

Figure 8. This shows the tree in figure 7 after wire-width tuning to achieve lower skew.

2.7 Tunable Grid-Trees

The examples above were all experiments or simple test cases. In this example we visualize an actual product clock distribution strategy which is a combination of tunable trees, all driving a single grid. This strategy combines many of the advantages of both grid and tree strategies, and has been used on a number of server microprocessors [6]. Figure 9 shows one quarter of the network for a Power4 GHz microprocessor [7]. There are 16 buffers in the network shown in figure 9, and these appear as vertical lines because buffers have delay, but do not transport the signal in the X-Y plane.

Interactive exploration of static 3D visualizations can be useful, or similarly an animation can be created using a programmed "fly-through". This is a third type of animation, where subsequent frames represent neither circuit time, nor design/tuning time, but simply show different viewing perspectives of static data.

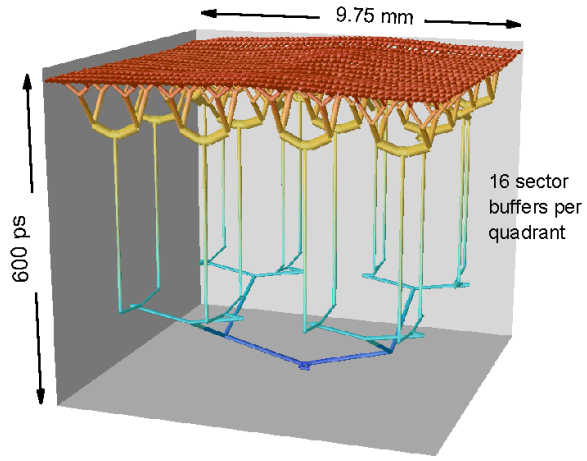

Figure 9. An optimized combination of tunable trees driving a single grid that has been used for several server microprocessors.

2.8 Power-Grid Noise

Another type of visualization is useful for understanding the power supply noise from chip and package transient simulations. Figure 10 shows one frame of an animation showing the complex interaction of circuit currents, capacitances and package inductance [8]. The lowest 2 points were found to be erroneous.

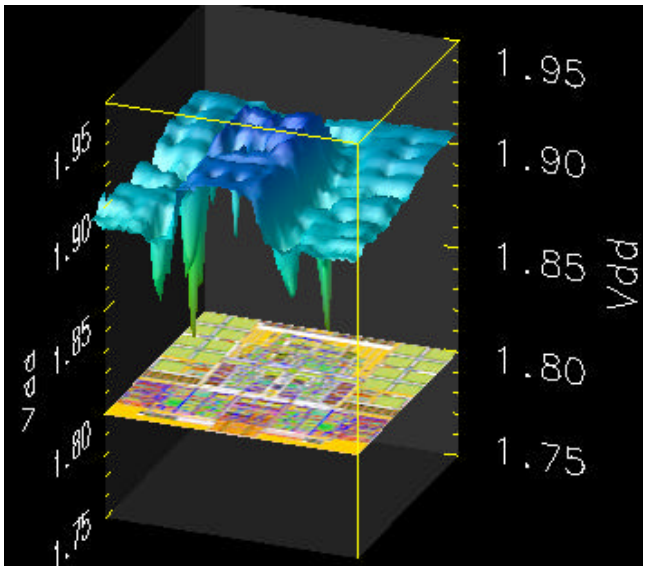

Figure 10. A single time-step of a power supply noise simulation on a microprocessor.

2.9 Two-Dimensional Layout Rendering

Although 3D visualizations often produce more striking images and animations, in practice 2D visualizations are often more clear, precise, and immediately useful. For example familiar 2D layout editors can have a number of "layers" that are not used for physical mask generation but are used instead to annotate and document the physical design layouts.

In actual design work, clever use of such annotation or highlight layers is probably the most common and effective visualization method used. Figure 11 shows a simple tree layout with skew annotated using crossed boxes. The authors routinely uses different colored annotations simultaneously in annotated layouts to visualize skew, slew, and capacitive loads.

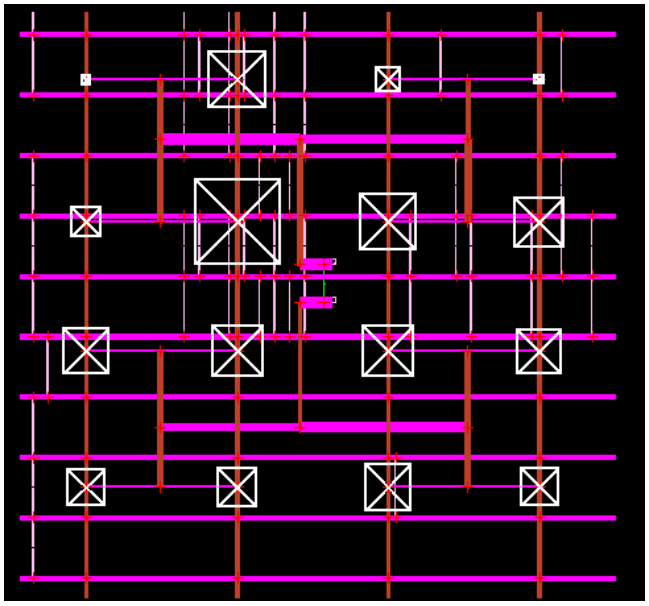

Figure 11. Layout view with skew shown using crossed squares.

Techniques such as this blur the artificial distinction between layout and electrical views, and can allow an extremely quick and efficient "feed-back loop" from the extracted simulations back to the actual layout. These relatively crude but quick visualization methods are actually extremely important for design-space exploration, manual tuning, checking, and communication. This suggests that the most valuable visualization methods might come from expanded annotation capabilities of VLSI design tools themselves. At a minimum, easy cross-probing capability between visualization tools and design tools is very important.

2.10 Two-Dimensional Full-Chip Pictures

Figure 12 shows a pseudo-photograph made from all the physical design layout data for a microprocessor [9].

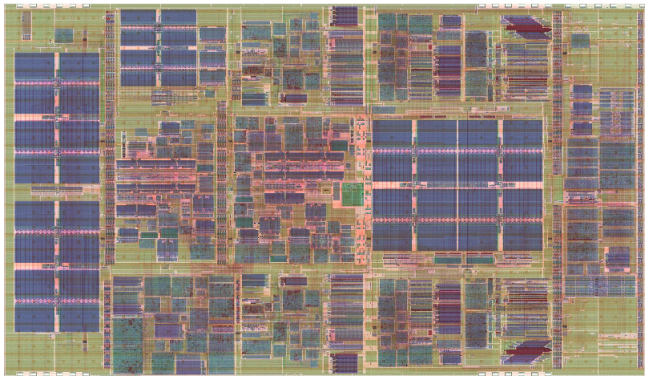

Figure 12. Pseudo photograph of microprocessor visulizing all design layers.

Years ago, chip photographs provided a useful basis for floor-planning discussions. With more metal layers, photographs have

become less useful, and must wait till hardware is available. This image was created using an optical attenuation model where the optical density is an exponential function of the density for each pixel [10]. This type of visualization has been used in place of chip micrographs since 1996 for IBM server microprocessors. Such an image was also used as the "floor" in figure 10.

2.11 Surface Plot of Wire Congestion

Another method employed for studying layout data is shown in figure 13, a frame from another animation. In this case, the Z-axis and color are determined by wiring density on the lower wiring levels of a microprocessor [7].

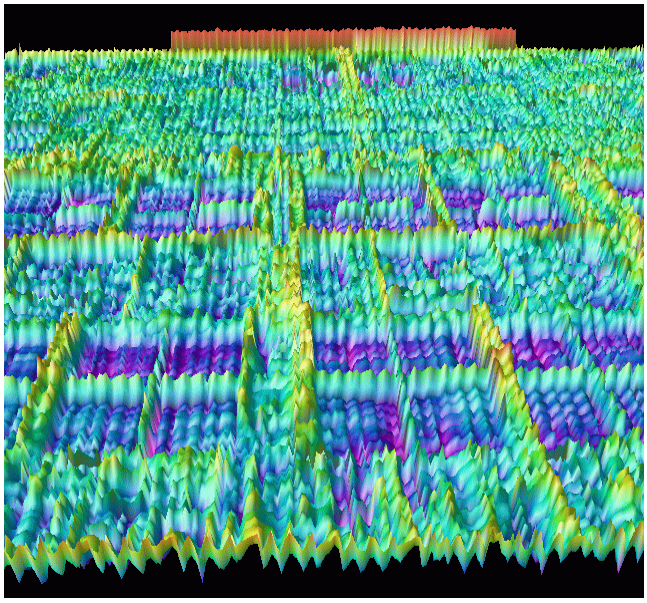

Figure 13. A Frame from an instruction-path fly-through of the POWER4 microprocessor lower-level wiring congestion.

2.12 Static Circuit Optimization

Visualizing a static circuit tuning process [11] required difficult visualization choices because the concepts being visualized are more abstract. As used previously, the Z-axis was used to represent delay. The elements being visualized are path delays or timing arcs through gates. Each cone-shaped object represents a path delay through a circuit block (in most cases simple gates). The width of each cone's base is the gate width driving that transition. Cones are bent to represent how sensitive each delay is to the gate size. Thus small-based cones tend to have more curvature, since the delay is sensitive to a small gate-size change.

Rising and falling delays may be different, and are thus represented by different cones. All the delay paths that can produce a single output transition are grouped so that the points of the cones have the same X and Y coordinates. The bases for each such group are then spread out a small distance from this point. For example, a 2-input NAND gate has 4 timing arcs (2 inputs, rising or falling) and thus produces two pairs of cones in the visualization. One pair of cones (whose points meet creating an inverted V-shape) represent the rising transition at the output, and one pair represent the falling transition.

All other visualizations in this paper simply used actual chip X and Y coordinates for the visualization's X and Y coordinates. In circuit tuning, however, the transistors may not have been placed yet, so physical X-Y coordinates are not available, and aren't very relevant anyway. Instead, one coordinate (X) is used to represent the number of gate levels between a particular gate and the circuit output pins. Although not particularly satisfying, the Y-axis represents how often each delay is in a critical path summed over all tuning iterations. Gates in the critical path are also colored darker (i.e. red). Figure 14 shows a small circuit before circuit tuning, and figure 15 shows the tuned circuit. These are the first and last frames in an animation representing the tuning process. Figure 16 shows a more realistically sized circuit, actually tuned and used on a microprocessor [5].

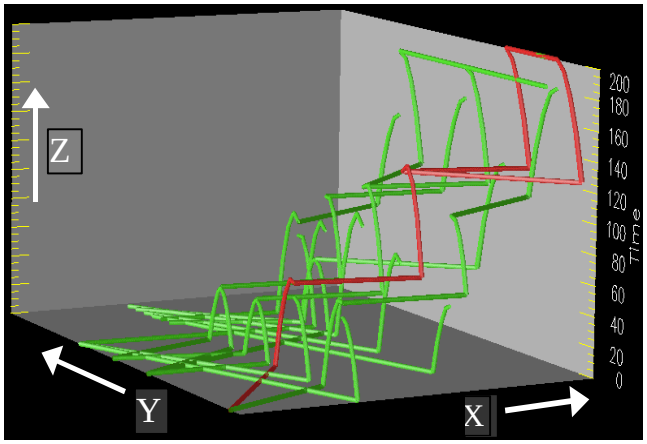

Figure 14. A circuit tuning visualization, before tuning.

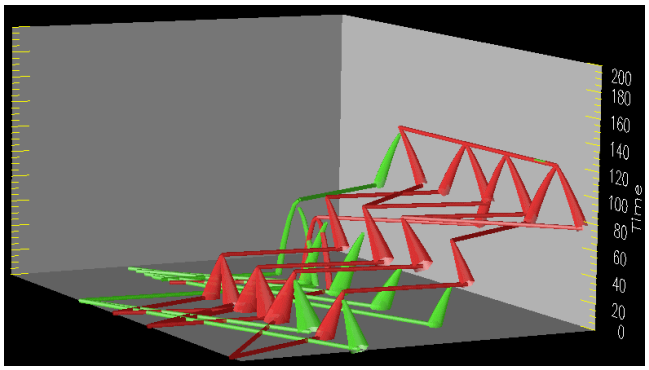

Figure 15. After Circuit Tuning.

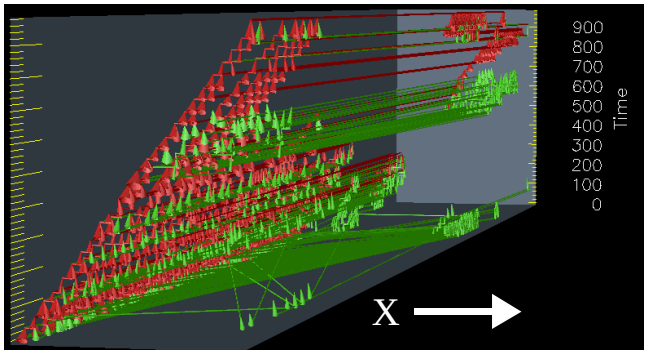

Figure 16. A more realistically sized tuning problem

In figure 15 we see that gate sizes increased (shown as wider cones), while decreases were obtained in delays (height)

sensitivities (curvature) and the maximum circuit delay compared to figure 14. Note also that almost all paths are critical (dark, or red) in the final tuned result.

3. DISCUSSION

It has become much easier for engineers to apply sophisticated visualization techniques to understand a huge variety of VLSI design problems with available hardware and open-source software [9]. It is less trivial to choose and implement visualizations that obviously speed the design process or improve quality in predictable ways. In fact, the unpredictable or incidental benefits often outweigh the planned benefits.

The most common unexpected benefit is the discovery of subtle bugs in tools or designs that often become immediately obvious upon viewing a rich visualization.

Another large benefit comes from the educational value. Tutorials, demonstrations, conference talks, and class lectures all need to convey new concepts as quickly and clearly as possible, while details and numerical accuracy (which often not well represented by rich visualizations) are less important.

Viewers with less experience in a particular specialty tend to be more appreciative of new visualizations. Perhaps the fact that a neophyte can quickly pick up hard-won insights from a new visualization that traditionally required careful analysis and deep study can seem unfair or superficial to the experienced designer.

The entertainment value can result in both positive and negative effects, depending upon the perspectives and prejudices of viewers. There is perhaps an unsaid perception that real designers don't need such *crutches*, and pretty visualizations are a waste of time. As mentioned above, many of the more experienced technical leaders have little need or interest in such visualizations.

Finally, although none of the visualizations presented here were created explicitly to sell a product or even a specific idea, it is clear that such visualizations might be valuable for advertising and marketing purposes. Hopefully, the inevitable marketing applications can evolve synergistically with the original *grass-roots* technical motivations for these visualizations.

4. CONCLUSION

Because of the appeal of new visualizations, the ease with which they can be created using readily available software [12], their educational and communication-fostering value, in addition to the occasional key insights obtained, many new visualization techniques will be explored by VLSI designers in the next few years. The most successful visualization techniques will allow us to apply the impressive visual processing capabilities in our own heads to understand, debug, and improve our increasingly complex VLSI designs.

5. ACKNOWLEDGMENTS

I'd like to thank the following people for collaborating in these design and visualization projects, and for helping maintaining an environment where such enjoyable explorations are tolerated (and even occasionally appreciated): Scott Neely, Mary Lanzerotti, Chandramouli Visweswariah, Greg Northrop, Steve Walker, Albert Ruehli, Howard Chen, James Venuto, Balaram Sinharoy, Giovanni Fiorenza, David Webber, Craig Carter, including management Michael Rosenfield, and Thomas Bucelot. The author is also most grateful to the creators of IBM Open Visualization Data Explorer [12].

6. REFERENCES

[1] A. Deutsch, G. V. Kopcsay, P. Restle, *et al*, "When are Transmission-Line Effects Important for On Chip Interconnections?" *IEEE Trans. Microwave Theory Tech.* (USA) Vol. 45, No. 10, pt. 2, pp. 1836-46, Oct. 1997.

[2] P. J. Restle, K. A. Jenkins, A. Deutsch and P. W. Cook, "Measurement and Modeling of On-Chip Transmission-Line Effects in a 400 MHz Microprocessor," *IEEE Journal of Solid-State Circuits*, Vol. 33 No. 4, pp. 662-665, Apr. 1998.

[3] P. J. Restle, A. E. Ruehli, S. G. Walker, "Dealing with Inductance in High-Speed Chip Design", In *Proc. of the Design Automation Conf.*, pp 904-909, June 1999, New Orleans, LA.

[4] P. J. Restle, A. E. Ruehli, S. G. Walker, G. Papadopoulos, "Full-Wave PEEC Time-Domain Method for the Modeling of On-Chip Interconnects", *IEEE Trans. on Computer-Aided Design*, (in press).

[5] T. Xanthopoulos, *et al*, "The Design and Analysis of the Clock Distribution Network for a 1.2GHz Alpha Microprocessor", *IEEE ISSCC Tech. Dig.* pp. 232-233, Feb. 2001.

[6] P. J. Restle *et al*, "A Clock Distribution Network for Microprocessors", *Symposium on VLSI Circuits Digest of Technical Papers*, June 2000, pp. 184-187, Honolulu, HI

[7] C. Anderson *et al*, "Physical Design of a Fourth-Generation POWER GHz Microprocessor", *IEEE ISSCC Tech. Dig.* pp. 232-233, Feb. 2001.

[8] J. Neely *et al*, "CPAM: A Common Power Analysis Methodology for High-Performance VLSI Design", *IEEE Proc. 9th Topical Meeting on Electric. Perform. Electr. Packag*, pp.303-306, Oct. 2000.

[9] Brian Curran *et al*, "A 1.1GHz First 64b Generation Z900 Microprocessor", *IEEE ISSCC Tech. Dig.* pp. 238-239, Feb. 2001.

[10] Gregory A. Northrop, IBM Research, *private communication*.

[11] C. Visweswariah, "Formal Static Optimization of High-Performance Digital Circuits", *Proc. Tau*, pg. 51, Dec. 2000.

[12] http://www.research.ibm.com/dx
http://www.opendx.org/

[13] http://www.research.ibm.com/people/r/restle