

Modeling and Simulation of Fiber Image Guide Multi-Chip Modules for MOEMS Applications

Steven P. Levitan¹, Timothy P. Kurzweg², Jose A. Martinez¹, Mark Kahrs¹, Jason Bakos³, Craig Windish¹, Jason Boles¹, John Hansson, Michael Wiesser⁴, Charles Kuznia⁵,
Donald M. Chiarulli³

¹Dept. EE, University of Pittsburgh, Pittsburgh, PA, 15261

²Dept. ECE, Drexel University, Philadelphia, PA 19104

³Dept. CS, University of Pittsburgh, Pittsburgh, PA, 15260

⁴Schott Fiber Optics, 122 Charlton Street, Southbridge, MA 01550-1960

⁵Peregrine Semiconductors, 6175 Nancy Ridge Dr., San Diego, CA 92121

ABSTRACT

Densely integrated systems in the future will incorporate device and communication technologies that span the domains of digital and analog electronics, optics, micro-mechanics, and micro-fluidics. Given the fundamental differences in substrate materials, feature scale and processing requirements between integrated devices in these domains, it is likely that multi-chip, system-in-package, integration solutions will be required for the foreseeable future. The multi-domain nature of these systems necessitates design tools that span multiple energy domains, time and length scales, as well as abstraction levels. This paper describes a case study of the modeling of a photonic/multi-technology system based on a 3D volumetric packaging technology implemented with Fiber Image Guide (FIG) based technology. It is 64x64 fiber crossbar switch implementation using three Silicon-on-Sapphire mixed signal switch die with flip-chip bonded VCSEL and detector arrays. We show a single end-to-end system simulation of the O/E crossbar working across the domains of free-space and guided wave optical propagation, GaAs O/E and E/O devices, analog drivers and receivers and integrated digital control.

INTRODUCTION

The case study presented in this paper is a demonstration of our mixed-signal, system level simulation tool Chatoyant. We have developed Chatoyant to support modeling and simulating of micro-opto-electro-mechanical systems [1][2]. Chatoyant is built upon the object-oriented simulation framework Ptolemy [3]. Chatoyant's component models are written in C++ with sets of user-defined parameters for the characteristics of each component instance. As shown in Figure 1, each icon represents a component model; while, each line represents a signal path (optical, mechanical, or electrical) connecting the outputs of one component to the inputs of the next. To maximize our modeling flexibility, our signals are composite types, representing the attributes of force, displacement, velocity, and acceleration for mechanical signals, voltages and impedances for electronic signals, and wavefront, phase, orientation, and intensity for optical signals. The composite type is extensible, allowing us to add new signal characteristics as needed. Optical propagation models are based on two techniques: Gaussian and diffractive scalar. Gaussian models give fast and accurate results for macro-scale systems and systems that exhibit limited diffraction, while more computationally intensive scalar models are used when diffraction effects dominate the system.

Chatoyant performs static simulations to analyze such effects as mechanical tolerancing, power loss, insertion loss, and crosstalk, while dynamic simulations analyze data streams with techniques such as noise analysis and BER calculation. In the following sections we discuss the philosophy and methodology used in the implementation of Chatoyant to support the simulation of multi-domain micro systems.

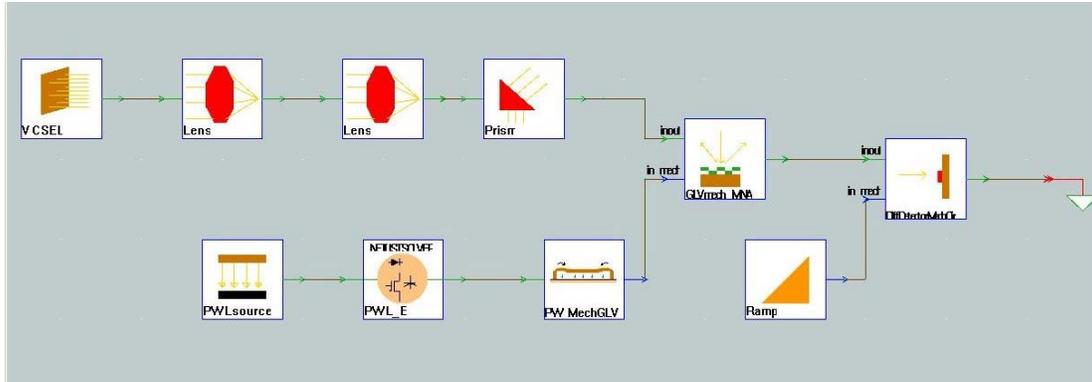


Figure 1: Micro optical system modeled in Chatoyant showing: VCSEL source, focusing system, prism, MEM switch controlled by a voltage amplifier, and positional detector

System Level Simulation

The simulation of multi-domain systems involves signals with different properties (e.g., voltage for electronics, force for mechanics and intensity for optics) and with varied dynamics. In Chatoyant, the use of an object-oriented framework permits a large degree of abstraction and flexibility for the simulation of such systems. At the highest level, the system is composed of component modules that are individually characterized and joined together by the mutual exchange of information. While in Figure 1 we show the modules as icons in the user interface, in Figure 2(a) we show how each module, i , processes some vector of input messages, $X_i(t)$, updates its vector of internal state variables, $S_i(t)$, and generates sets of output messages. Using a discrete event (DE) simulator, each module's execution is based on the availability of new data values for its inputs. The simulation scheduler provides the simulator with a buffering capability, which allows it to keep track of all the messages arriving at one module when multiple input streams of data are involved. This provides for the modeling of dynamic systems where each component can have variable rates of consumed or produced data during simulation.

In general, the components come from a parameterized model library. Some examples include CMOS analog amplifiers, vertical cavity surface emitting lasers (VCSELs), micro-mechanical cantilevers, lenses, and moving mirrors. The components are modeled at the behavioral level where they are represented either by analytic expressions or as a tightly coupled network of elements such as shown in Figure 2(b). In either case, at the system level there is a loosely coupled network of tightly coupled component models. This corresponds well with the general structure of mixed-signal microsystems where multi-domain components interact with few signals, while at the same time, the behavior of each component is based on its underlying physical processes.

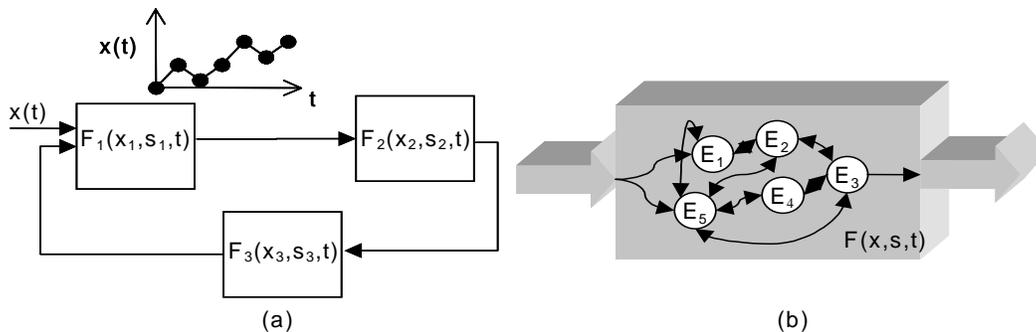


Figure 2: System modeling methodology (a) Inter-component; and (b) Intra-component interactions

Behavioral Modeling Methodology

Whereas at the system level each component is a black box, each component can be described as a network of linear and non-linear elements. For our modeling methodology, the component is first decomposed into a nodal representation of elements, as shown in Figure 2(b), where elements are interconnected through nodes.

We perform linear and non-linear sub-block decomposition of the circuit model of the device, as shown in Figure 3(a). This decomposes the design into a linear multi-port sub-block section and non-linear sub-blocks. The linear multi-port sub-block can be thought of as characterizing the interconnection network and parasitic elements while the non-linear sub-blocks characterize elements with active non-linear behaviors.

In the second step, a Modified Nodal Analysis (MNA) is used to create a mathematical representation for the device, shown in Figure 3(b). In this expression $[S]$ is the storage element matrix, $[G]$ is the conductance matrix, $[x]$ is the vector of state variables, $[B]$ is a connectivity matrix, $[u]$ is the excitation vector, and $[I]$ is the current vector [4].

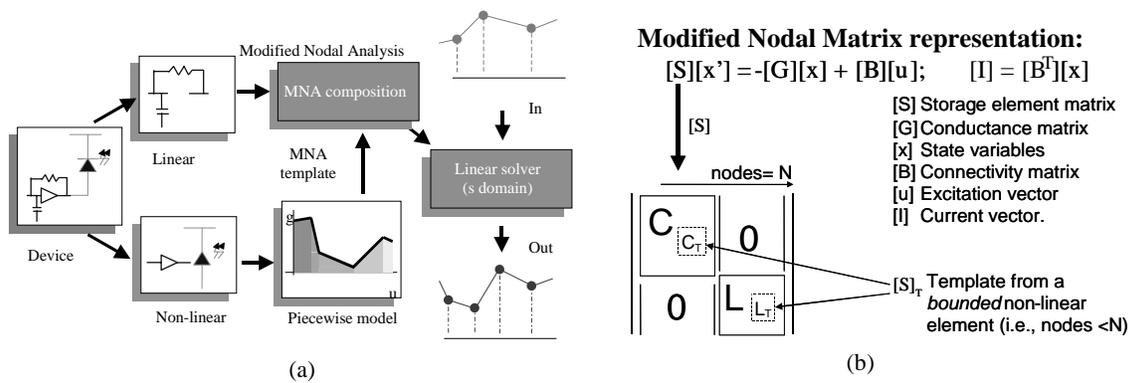


Figure 3: (a) Piecewise modeling for MSMD devices (b) MNA format and template integration

The linear sub-block elements are directly mapped into this representation, but the non-linear elements need to first undergo a further transformation. We perform piecewise modeling for each non-linear sub-block. This technique, as explained in the next section, translates the structure and behavior of these elements into the form of placeholders for their MNA templates. The templates give us the ability to change models for the non-linear devices based on changes in conditions in the circuit, and thus the regions of operation of each element. The templates generated can then be integrated to the general MNA containing the linear components adding their matrix contents to their corresponding counterparts. This process is shown in Figure 3(b) for the storage, or memory, matrix $[S]$. Once the integrated MNA is formed, a linear analysis can be performed to obtain the solution of the system. Additionally constraining the signals in the system to be piecewise linear in nature allows us to use a simple transformation to the time domain without the use of costly numerical integration.

During each time step in the simulation, the state variables in the module will change and might cause the non-linear elements to change regions of operation. Therefore, we re-compute the solution caused by changes between piecewise models. In general, depending on the number of regions of operation used in the piecewise linear model, there are a large number of time steps during which the system representation is unchanged, justifying the computational savings of this technique. Understanding that the degree of accuracy of piecewise linear models depends strongly on the step size chosen for the time base, an adaptive control method is used. The inclusion of the samples during fast transitions or suppression of time-points during "steady state" periods optimizes the number of events used in the simulation.

While we have used the electrical MNA representation for the preceding discussion, these same techniques apply to other domains. For mechanical components, we can derive a similar template based structure for composition into a MNA formulation. With damping forces proportional to the velocity, the equation of motion for a mechanical

structure with viscous damping effects is: $F = KU + BU' + MU''$ [5] where, K is the stiffness matrix, U is the displacement vector, B is the damping matrix, U' is the velocity vector, M is the mass matrix, U'' is the acceleration vector, and F is the vector of external forces affecting the structure. Similar to the previous case, this equation represents a set of linear ODEs if the characteristic matrices K , B , and M are static and independent of the dynamics in the body. If the matrixes are not static and independent (e.g., the case of aerodynamic load effects), they represent a set of non-linear ODEs.

Using a modification of Duncan's reduction technique for vibration analysis in damped structural systems [6], we reduce the above general mechanical motion equation to a standard first order form, similar to the one discussed in the previous section which gives a complete characterization of a mechanical system, as shown in Fig. 7(b). Each mechanical element (beam, plate, etc.) is characterized by a template consisting of the set of matrices Mb and Mk , composed of matrices B , M , and K . If the dimensional displacements are constrained to be small and the shear deformations are ignored, the derivation of Mb and Mk is simplified and independent of the state variables in the system. Typically, this element is only a part of a bigger device made from individual components that are characterized using similar expressions. The generalization of the previous case to an assembly of elements or mechanical structures is fairly straightforward.

We use dynamic control of the sampling rate in the mechanical domain based in the Nyquist criteria of the highest significant modal frequency for the structure. The allowed sampling rate is lower than half of the period of the highest modal frequency. This allows us to optimize the samples used in this domain while still completely characterizing its dynamic behavior. There can be several orders of magnitude reduction in the sampling rate compared to the electrical domain because of the difference in dynamics.

As mentioned above, our behavioral modeling technique is based on the ideas of nodal analysis together with the use of piecewise linear models for non-linear behaviors, we present the underlying technique for these methods next.

Piecewise Linear Modeling

The nodal analysis principle can be traced back to the basic conservation laws of energy and bond graph theory []. In an enclosed volume with finite interfaces, an energy conservation relationship can be established using the energy flow through the interfaces and the internal energy density. The behavior of each element is captured in terms of the analytic relationships among variables, which define the state of its nodes. The two basic types of variables in nodal analysis are *across* and *through* variables. Across variables are measures of the values of field potential in the physics of the device (e.g., Electrical potential, Temperature, Fluid Pressure). Through variables are measures of flux intensity at nodes (e.g., Electrical current, Thermal flux, Fluid velocity).

Consequently, for any element in the nodal representation a function f can be found that relates the total flows (across variables) through its interfaces (nodes) as equal to zero. For use in our simulation methodology, we need a way to provide a linear approximation of $f_i(\hat{s})$ for each of the i ($1 < i < m$) nodes that make up the ports of the element. In general, this will be a hyperplane in the dimensionality of the domain of s , plus one for the range of f . While there are many choices for the decomposition, the resulting piecewise linear approximation of the function should have the following characteristics: it must be linear in the dimensionality of the function (e.g. planes in three-space for functions of two variables); it should approximate the function within specified absolute and relative tolerances; it should make the fewest partitions possible; and, it must be mathematically continuous in the underlying function value at the transition points between the different regions of operation. This last point reduces the probability that the device model, in simulation, will oscillate between regions of operation due to poor convergence. It also makes simple curve-fitting, (e.g., based on minimizing RMS error) techniques less applicable. Rather, we use a triangulation approach based on recursive decomposition of the function space.

We define a $k+1$ dimensional space for the domains of the k independent variables of s and the range of f . We recursively decompose the k dimensional projection, from the independent variables, of the space into k dimensional hypercubes. The decomposition need not be symmetric; we can refine one region of the space more than others. Each hypercube defines the region of operation for the element. The advantage of cubic decomposition is that the

bounds for each region of operation can be tested efficiently in the simulator. Recursion stops when we reach our tolerance level in every region.

The next step is to triangulate the hypercubes, since we want a linear approximation of the function. For functions of more than two variables, the triangulation of the resulting hyper-cubes is not simple. While, the best decomposition of the 3D cube is 5 tetrahedra and 4D hypercubes decompose in to 16 hyper-tetrahedra, in higher dimensions the decomposition grows to be quite large and finding the optimal is a very difficult problem. In this work we have used a straightforward vertex index permutation approach.

Figure 4(a) and (b) illustrate the case for three dimensions. After recursive decomposition of the space into cubes, each cube (in Figure 4(a)) represents an interval on the domain for a function of three variables. In Figure 4(b), we show the tetrahedral triangulation of a single 3D cube, where each tetrahedron is a linear approximation of the function. As an example of this method, in Figure 4 (c) we show the 2D linearization of the simple NMOS transistor equation [8] to %1 relative accuracy.

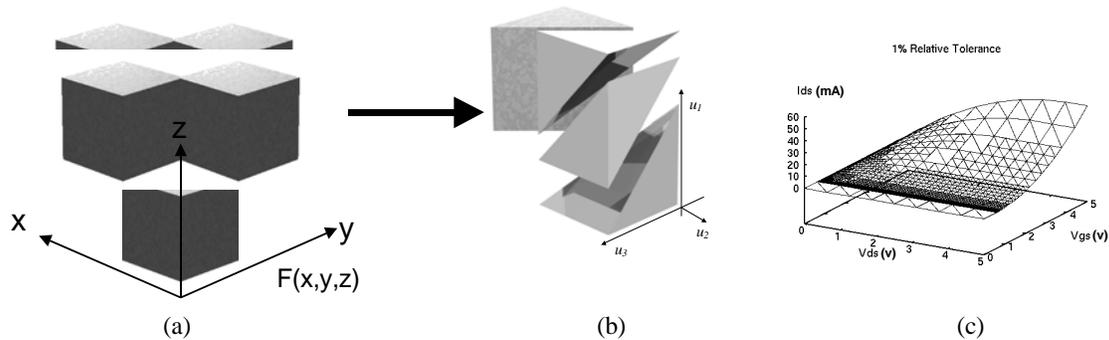


Figure 4: a) Function $F(x,y,z)$ is recursively decomposed in hypercubes b) Decomposition of cube into six tetrahedrons representing a piecewise linear function in 3-space c) . 2D Linearization of NMOS transistor, I_{ds} vs. V_{gs} and V_{ds} for 1% relative accuracy

This coupled modeling methodology allows us to directly translate the interaction of multiple domains in the same simulation framework without any cumbersome transformation. Its applicability however, is clearly dependent on the availability of the behavioral function for the underlying physical system.

Digital and Analog Co-Simulation

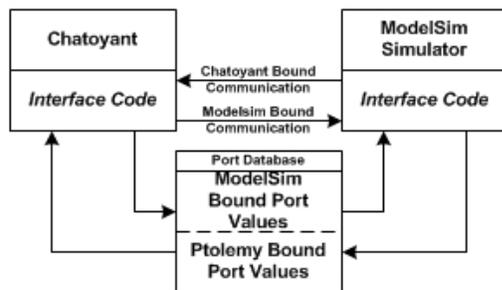


Figure 5 Chatoyant – Modelsim Co-simulation

Chatoyant is built on top of Ptolemy (Classic 0.71) developed at University of California, Berkeley, and uses both discrete event (DE) and dynamic data flow (DDF) simulation methods. However, neither Chatoyant nor Ptolemy provides library components or features that can directly simulate digital design circuits in the form of RTL or behavioral designs written in VHDL or Verilog. To handle these components, we have chosen to use the commercially available ModelSim HDL simulator for the components of a whole system that require an HDL simulator [9].

Figure 5 shows the architecture of the co-simulation environment. The implementation of this environment must address two issues: First, how is information exchanged

accurately between a digital domain simulator (with ML19 conventions) and an analog simulator? Second, how is synchronization performed for events and data exchange between these two simulators? The first issue can be addressed by the means of technology dependent lookup tables, converting between symbolic digital values and analog voltages and impedances.

Since both simulators are discrete event simulators, the second issue can be solved with parallel discrete event simulation (PDES) techniques [10]. In PDES, there are two fundamental synchronization approaches, classed either as conservative or optimistic approaches. Conservative approaches require all simulators to remain synchronized, never simulating to a future time until all simulators are ready to proceed. Optimistic approaches, on the other hand, allow any particular simulator to simulate beyond a certain point, without requiring other simulators to be synchronized. However, if an event comes to a simulator that is in the past, relative to its own local time, then that simulator must “rollback” to a state that is before the incoming event’s time.

The conservative approach gives a solution requiring less memory. By ensuring that both simulators (one being Chatoyant and the other being ModelSim) be consistently synchronized this becomes a matter of only passing event information between the simulators. Therefore, we have developed an event look-ahead method where each simulator event queue is examined and the simulation interval is calculated based on the next event time in each of the simulator’s respective event queues. This avoids stepping through long periods of null events. The algorithm gives results as accurate as each simulator would individually, since synchronization occurs exactly when it needs to.

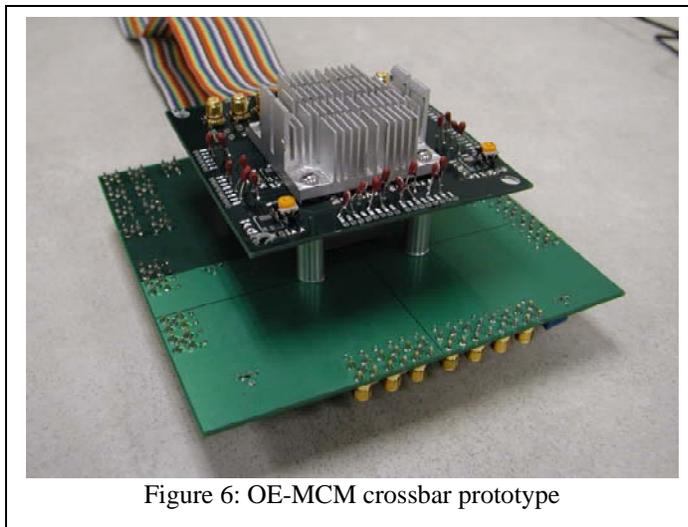


Figure 6: OE-MCM crossbar prototype

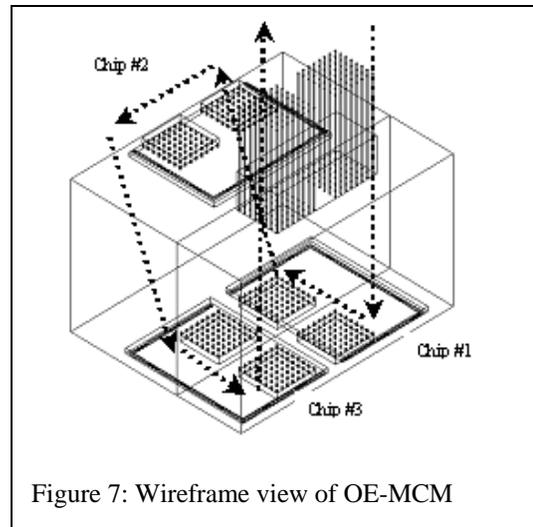
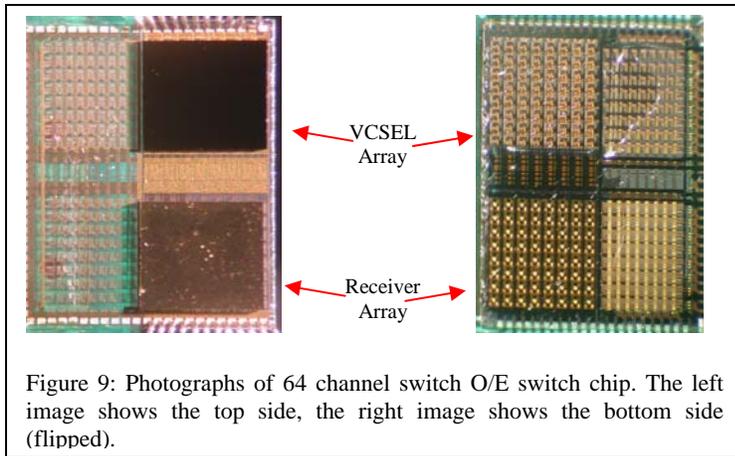


Figure 7: Wireframe view of OE-MCM

OE-MCM CROSSBAR SWITCH DEMONSTRATION SYSTEM

Our demonstrator system consists of a 3-chip OE-MCM implementation of a 64 channel switch fabric [11] [12]. A photograph of the prototype is shown in figure 6. The OE-MCM is positioned between the two PCBs with optical fiber routed into and out the switch vertically through an opening in the heat sink. At the core of the MCM architecture is an optical element built by bonding together two rigid segments of imaging fiber guide. These image guides are produced by Schott Fiber Optics and consist of a dense array of small core fibers arranged in a lattice [13]. Fiber diameters typically range from 5 to 20 microns, yielding core densities of two thousand to fifteen thousand cores per square millimeter. Thus, an array of optical channels imaged on one surface is correspondingly imaged on the opposite surface. It is important to keep in mind that this is an imaging operation. Each optical channel is spatially over-sampled by multiple fiber cores. This is not a core-per-channel arrangement such as you have in a fiber ribbon cable.

Figure 7 is a wireframe diagram showing the physical structure of the OE-MCM and the optical paths taken by signals moving through the switch. OE-chips are directly bonded to the end surface of the fiber bundle such that



direction.

The OE chips are shown in figure 9. Each of OE chips implements eight independent 8x8 switching elements. The three chips are connected in a 3 stage non-blocking switch network. The chips measure 4mm x 5.5mm and were fabricated part of the Peregrine ultra-thin silicon-on-sapphire (UTSi) COOP run. Each switch chip implements 8, 8x8, switches in CMOS logic, as well as 64 channel (8x8) driver and receiver arrays. Switch configuration is done electrically with configuration information is stored in a two-level memory. The “map” level of the configuration memory drives the configuration of the switch logic. The “cache” level is a second copy of the configuration loaded externally through the electrical interface. To conserve I/O pins the cache level is loaded over multiple cycles of a 12-bit I/O bus. Once a new configuration is completely loaded into the cache, the switch configuration is set in a single parallel transfer between the cache and map level memory.

The optical elements on the OE-chips consist of separate 8x8 VCSEL and GaAs PIN photodetector arrays each mounted active side down using flip chip bonding. Thus the optical I/O is done through the transparent SoS substrate into and out of the bottom surface of the die. After the photodetector and VCSEL chips are bump bonded to the switch chips, the resulting OE chips were then bonded directly to an optical element built from two segments of rigid fiber image guides. Alignment marks were created using the top metal layer and placed throughout the chip to facilitate alignment when bonding the dies to the optic. The fiber bundles guide all of the optical signals between the chips without additional optical elements. Optical input and output is implemented with a pair of 2D (8x8) fiber ribbon cables that are also directly bonded to the image guide optic. Electrical connections are made via bump bonds between the UTSi devices and conventional printed circuit boards on which the MCM is mounted.

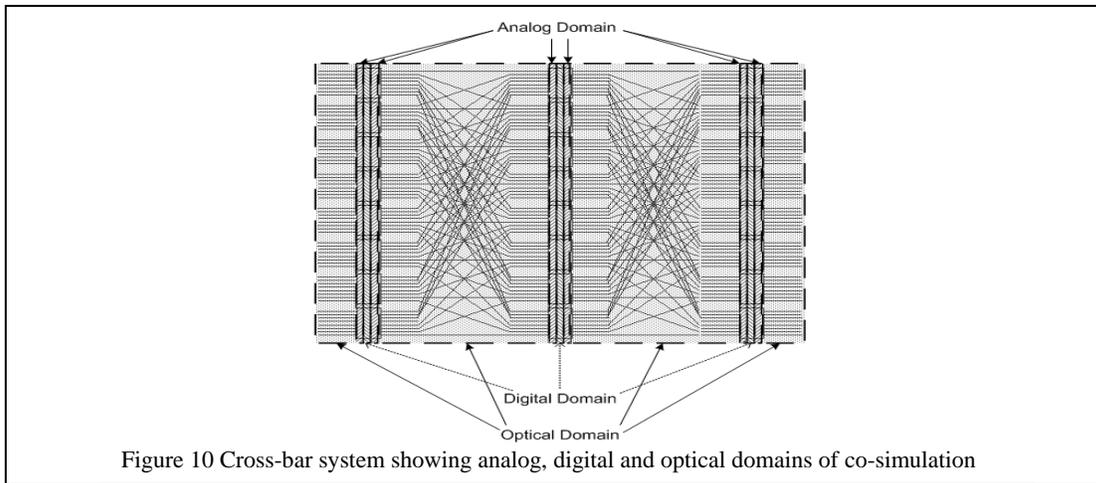
optical signals traverse to transparent silicon-on-sapphire substrate and are coupled into the fiber guide. Since the fiber bundle segments are rigid, the waveguides become both the structural elements and the communication channels. As can also be seen from figure 8 the optic is made from two separate segments of fiber image guide. The left half of the chip is on top of the section where the fibers run orthogonal to the chip surface. The right half of the chip is on top of the section where the fibers run 70 degree off the chip surface. This is shown in the bottom view of the fiber image guide. The side view of the optic shows the fiber



Figure 8: Images of optic with one switch chip mounted on the top, shown from the top, bottom, and side.

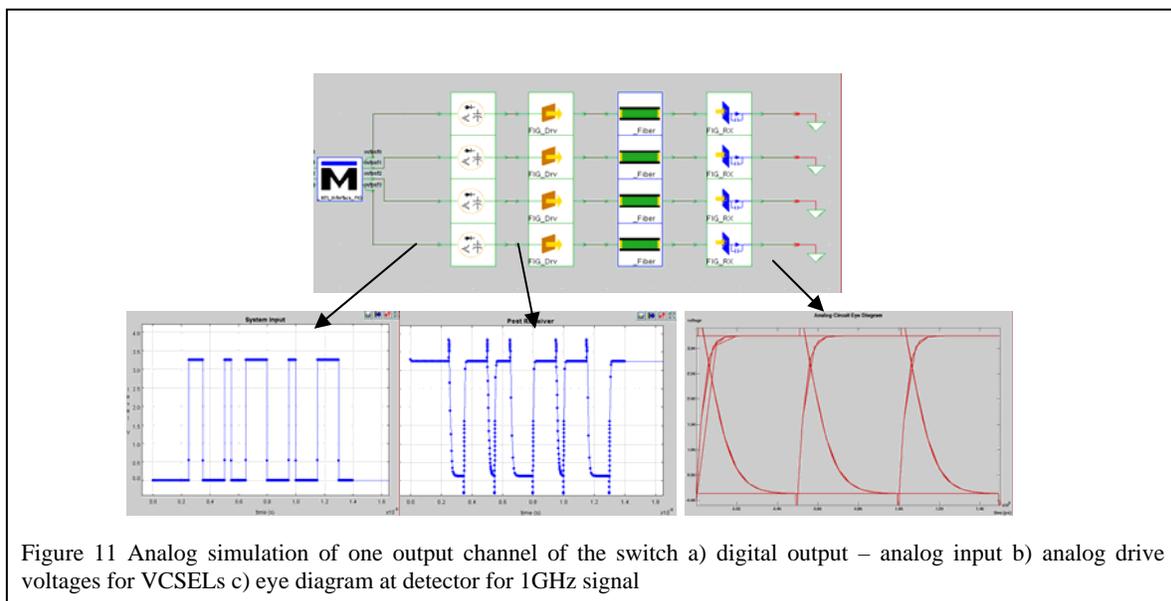
SIMULATION RESULTS

To illustrate the capabilities of the simulation environment, a system simulation of the optoelectronic crossbar switch was used. As discussed above, this system is composed of guided wave optics, and both optoelectronic and analog circuits (simulated in Chatoyant), and digital hardware (simulated in ModelSim). The information flow in the system consists of three stages of switches. At each stage, optical data streams are detected by photo-diodes, amplified in analog circuits, routed by digital circuitry, and sent through analog amplifiers to a vertical cavity surface emitting laser (VCSEL) array. Light is passed between stages by guided wave optics. Figure 10 shows the block layout of the system.



Analog Electrical Simulation

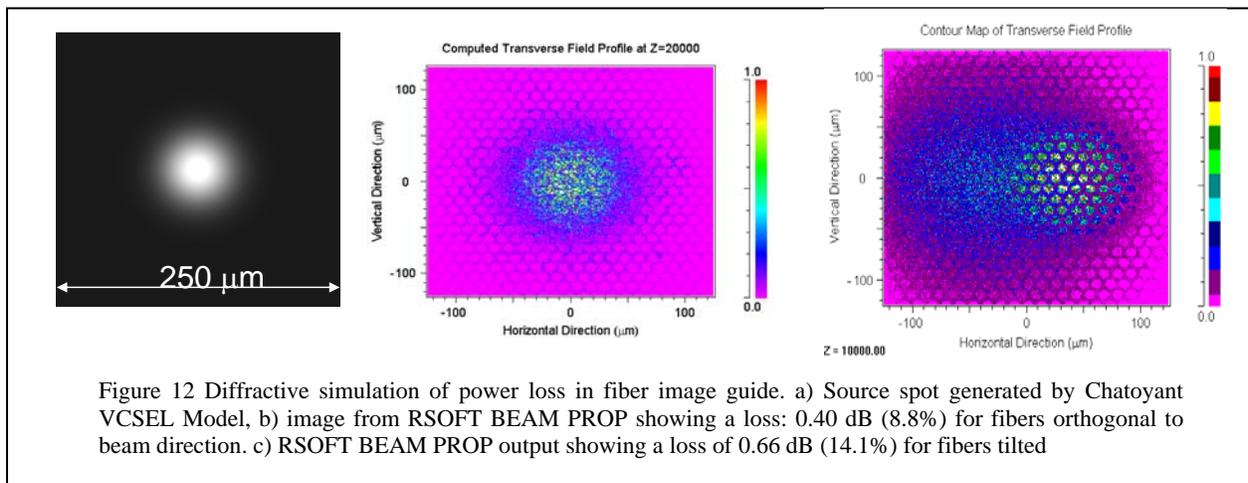
Figure 11 shows several stages of analog simulation based on the piecewise linear models for CMOS circuits built in the Peregrine Semiconductor UTSi silicon on sapphire 0.5um process. In this simulation, we take a digital waveform generated by the VHDL simulator and use it as input to the analog simulator in Chatoyant. where the signals are inverted and amplified to drive the VCSEL models. Optical signals from the VCSEL models are propagated and



used to drive the PIN photodiode models using Chatoyant models for optical propagation. Finally, we show an eye diagram at the output of the photodetectors for a 1GHz signal. This should be compared against the 500Mhz eye diagram for the fabricated system.

Diffractive Optical Simulation

The optoelectronic VCSEL models in Chatoyant generate a diffractive scalar wavefront. For freespace propagation, we use an angular spectrum technique [11]. However, for guided wave optics we provide an interface to the Beam Prop software from RSOF Inc. Figure 12 shows two simulations of a VCSEL spot propagating through the fiber image guide. For this image guide the core diameters were 9 μm ($N=1.8$) surrounded by a cladding ($N=1.52$) with a pitch of 12 μm . For the case that the fibers are aligned to the axis of propagation we see 8% loss after 1mm of fiber. For the case where the fibers are tilted relative to the optical axis we see 14% loss over the same distance.

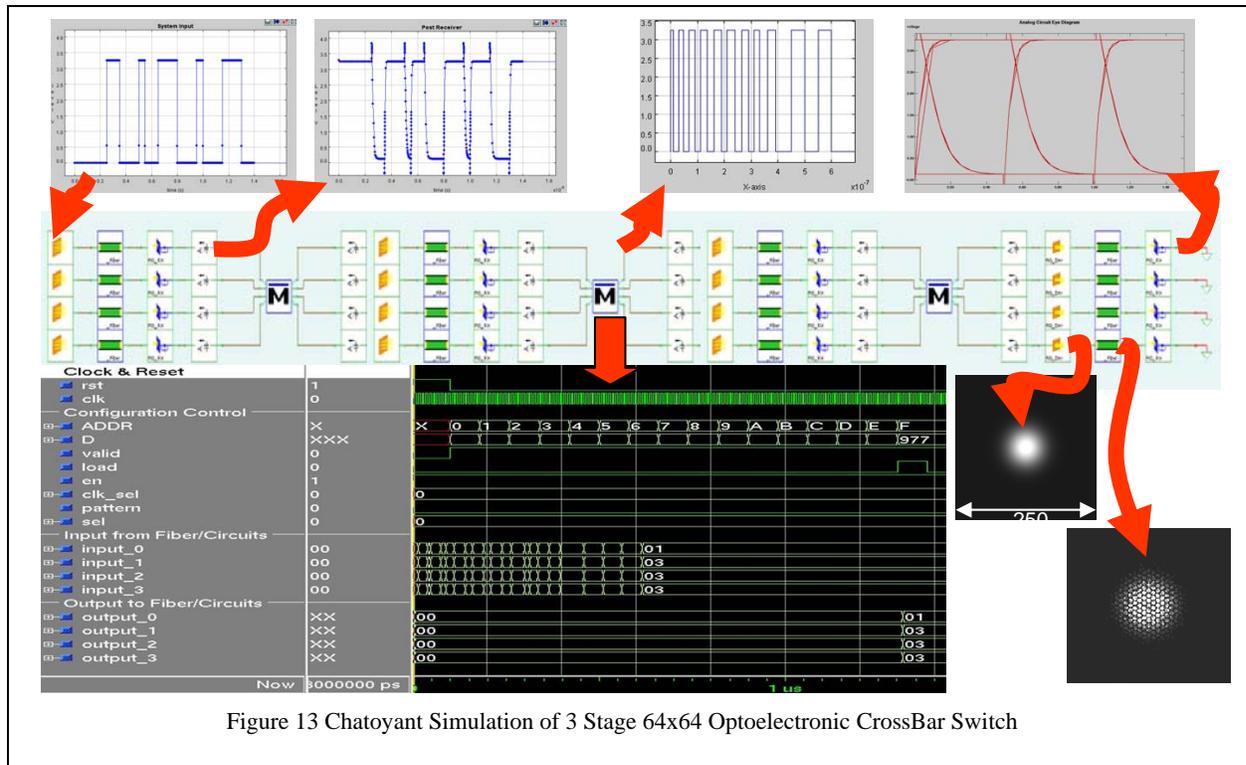


System Simulation

Figure 13 shows the simulation of this system. The representation of the system is shown in the middle of the figure. On the top starting at the left, one analog channel is shown entering the system, converted to an optical signal, passing through the FIG and being converted back to electronics. At the bottom of the figure, we show a co-simulation with ModelSim, a commercial VHDL simulator from ModelTech, which captures the digital behavior of the UTSi CMOS switches. The top right of the figure shows the signal passing out of ModelSim, converted to an analog signal, passing back through another stage of switches converted back to optics and then detected at the output of the system. The lower right portion of the figure shows intensity profiles of the optical signal passing through the FIG. The system was simulated for 1.3 μs of a 1GHz clock, on a 2.4GHz 1GB Pentium P4. The simulation time was 3.36 minutes with 8075 events.

SUMMARY AND CONCLUSIONS

In this paper we have shown both the need and utility of a mixed-signal multi-domain system level simulation tool for multi-chip optoelectronic systems. The need for such tools arises from the ever increasing levels of integration and the ever broadening space of technologies that are being incorporated into computing, control, and communications systems. The need for simulation tools that span the domains of analog and digital electronics as well as free-space and guided wave optics will continue to grow as optical MEMS and opto-bio-fluidics applications emerge as mainstream commercial systems.



REFERENCES

1. Levitan, S.P., Kurzweg, T.P., Marchand, P.J., Rempel, M.A., Chiarulli, D.M., Martinez, J.A, Bridgen, J.M., Fan, C., McCormick, F.B., "Chatoyant: a computer-aided design tool for free-space optoelectronic systems," *Applied Optics*, Vol. 37, No. 26, September, 1998, pp. 6078-6092.
2. Steven P. Levitan, et.al, "System Simulation of Mixed-Signal Multi-Domain Microsystems with With Piecewise Linear Models," *IEEE Trans. CAD*, 22 (2), pp.139-154, (February 2003).
3. J. Buck, S. Ha, E. A. Lee, and D. G. Messerschmitt, "Ptolemy: A framework for simulating and prototyping heterogeneous systems," *Int. J. Computer Simulation*, Vol. 4, 1994, pp. 155-182.
4. P. Feldmann, and R. W. Freund, "Efficient linear circuit analysis by Padé approximation via the Lanczos process," *IEEE Trans. Computer-Aided Design*, vol. 14, May. 1995, pp. 639-649.
5. J. S. Przemieniecki, *Theory of Matrix Structural Analysis*, McGraw-Hill, New York, New York, 1968.
6. W. J. Duncan, Reciprocation of Triply Partitioned Matrices, *J. Roy. Aeron. Soc.*, vol. 60, 1956, pp. 131-132.
7. D. L. Karnoop, and R. C. Rosenberg, *System dynamics: a unified approach*, John Wiley & Sons Inc., 1975.
8. A. S. Sedra, and K. C. Smith, *Microelectronic Circuits*, Fourth Edition, Oxford University Press, New York, 1998.
9. ModelSim SE Foreign Language Interface, Version 5.7d. 2003
10. Banerjee, Prithviraj. Parallel Algorithms for VLSI Computer-Aided Design. Prentice Hall. New Jersey. 1994.
11. Jason D. Bakos, Donald M. Chiarulli, and Steven P. Levitan, "Optoelectronic Multi-Chip-Module Implementation of a 64-Channel Fiber Switch," 2002 International Topical Meeting on Optics in Computing 2002. pp 161-163.
12. J. Bakos, D. Chiarulli, S. Levitan, "Optoelectronic Multi-Chip Module Demonstration System," *OSA Optics in Computing (OC 2003)*, OThD6, Washington, DC, June 18-20, 2003
13. Donald M. Chiarulli, et.al, "Demonstration of a Multichannel Optical Interconnection by use of Imaging Fiber Bundles Butt Coupled to Optoelectronic Circuits," *Applied Optics-IP*, Vol. 39 Issue 5 Page 698 (February 2000).
14. Timothy P Kurzweg, "Optical Propagation Methods for System-Level Modeling of Optical MEM Systems," Ph. D. Electrical Engineering, University of Pittsburgh, August 2002.