Computer-Aided Design of Free-Space Optoelectronic Interconnection (FSOI) Systems

S. P. Levitan[☩], P. J. Marchand[❖], M. A. Rempel[☩], D. M. Chiarulli[★], F. B. McCormick[❖]
Department of Electrical Engineering[☩] and Department of Computer Science[★]
University of Pittsburgh
Department of Electrical and Computer Engineering[❖]
University of California, San Diego

Abstract

New research in optoelectronic devices, which have made it practical to use optoelectronics in computing and communications systems, as well as the need for these systems to support higher information capacities has brought about a growing need for design and analysis tools for optoelectronic systems. While there are many research groups developing new and exciting optoelectronic devices, the integration of these devices into practical systems has been slow to follow. The reason for this lag is that researchers who design systems need to be able to evaluate how these new devices can be used to make components, and then how these components can be used to build systems. By having tools for effectively evaluating new designs based on new devices, system designers will be able to evaluate possible designs, and give feedback to materials and devices researchers for improved components.

Introduction

Free-Space Optoelectronic Interconnection Systems (FSOI) will become key components of the next generation of computers and communications networks. Prototypes of these systems have been proposed, designed and constructed for the last 20 years [1]. However, these systems have only existed in university and industry laboratories. To date, they have not seen general use. One of the reasons for this phenomena is that the time and effort involved in designing and building these systems, even as prototypes, is prohibitively expensive. Aside from some work in the area of CAD for fiber networks [2][3][4] these designs are currently performed essentially by hand. Therefore, the ability to make the kinds of design tradeoffs necessary for production-quality systems is lacking.

The current “state of the art” for these analyses is to first perform basic characterizations of the devices built in the laboratories and then to use those models and a set of ad-hoc procedures to generate end-to-end system performance estimates. This painstaking technique results in rough approximations which must be refined by actually prototyping each of the particular systems under consideration. The result is that few FSOI systems have been designed, and fewer still have been built. This is in sharp contrast to the growth of rapid prototyping systems in the electronic domain, where the path from concept to system is often as short as a few weeks.

In this paper we define the requirements for a true computer aided design system for hybrid optoelectronic information processing systems. There are three steps required to accomplish this goal. First, we must define the appropriate levels of abstraction for optoelectronic systems, analogous with the behavioral, logical, and electrical abstraction levels associated with digital electronic design. Second, we must characterize the necessary simulation models for these levels of abstraction. Third, we must specify the requirements for a multi-level simulation system, which can use these models to perform the analysis required to close the synthesis/analysis design loop.

One question which we must answer is: Can a single system support the various kinds of design that take place using very different system models, and implementation domains? The design of optoelectronic systems span the domains of free-space optics, optical fiber systems, integrated optics, as well as, high speed analog/digital electronics. A second question is: Can a single system support the multiple design tasks required for system level design? This includes the traditional functional design of systems, as well as the physical design of three dimensional hardware and the component design of electronic and optical devices. Our claim is that an integrated *system* design tool is both desirable and possible. We explain what we mean by a system tool in the next sections.

Optoelectronic System Design

The design space for high performance systems of any kind is very multidisciplinary and the number of disciplines required to build optimal systems is growing. This is evident in high performance electronic computing and switching systems, which must incorporate analog and digital electronic design, advanced software and operating system techniques, and leading-edge packaging concepts to reach their performance goals. If FSOI systems are to be competitive in such an arena, this trend will become increasingly apparent. In addition to all of the disciplines required by the electronic systems, FSOI systems will also require design work in conventional, micro, and waveguide optics, optomechanical systems, optoelectronic components, etc. Such a multidimensional design space begs for a tool to provide a coherent high-level picture of the design process and to enable a "holistic" analysis of the system-level trade-offs and the impact of these trade-offs on the component requirements.

We can consider the process of optoelectronic systems design from our experience with CAD for electronic systems. Design is an iterative process of synthesis, or creation, followed by analysis, or evaluation with each iteration of this synthesis/analysis loop expanding pieces of the final design into a "design hierarchy" of components or sub-systems. The design of optoelectronic systems must additionally include explicit input/output analysis of the components at each level of the decomposition hierarchy as well as defining the technology that would be used for each component at that level. After those decisions are made, the components themselves can be refined. One view of the functionality space for such a CAD system is shown below in Table 1. It incorporates a number of design disciplines, as well as examples of some of the design tasks required at different levels of design.

Therefore, to support optoelectronic systems design we must define the appropriate abstractions or "views" of optoelectronic systems, provide models for sub-systems in terms of those abstractions, and create analysis tools which use those models to help the designer perform the trade-offs, optimizations, and technology choices necessary to produce high-quality systems.

Abstraction

Figure 1 shows a "Basic Block" as an abstraction for general optoelectronic components. These will be the basis of our abstraction mechanism. While this kind of black box model is very simplistic, it has several advantages. First, it is general enough to be used for decomposition. In other words, it is the analog of the "module" in digital electronic design. Second, it encompasses electronic, optoelectronic and optical components with the same abstraction. This allows the system designer to use a black box approach until he or she is ready to decide on the appropriate technology for each component of the system. Third, it explicitly models the electrical and optical *signal transformation*, of power, noise and physical attributes of the components. While this is often ignored in the early design phases of electronic systems, it is an essential aspect of many optoelectronic systems. As discussed below, in optics we also need to explicitly model *signal propagation* as well. By explicitly modeling propagation, we encompass the issues of dispersion, cross talk, noise, etc., which are essential for the design of a free space system.

As shown in Figure 1, each component in the system is modeled as an object which could take its inputs in either electronic or optical form, and generate its outputs in either form as well. Power for amplification or modulation could also be either electronic or optical. Clearly for most components, some of these paths would only exist in one form. For instance, a continuous laser source would only have an electronic power source and an optical output. While a spatial light modulator could have electronic control inputs, optical data inputs, and optical data outputs, as well as an electronic power input.

Table 1. Functionality Space of OE CAD Systems

Electronics	Optoelectronics	Optics	Thermal	Packaging/Mechanics
Functional, Models	Analytic models	Lens law, Image formation	Power density	Area, Volume
Logic, Timing		1st order layout, Paraxial Gaussian beam propagation	1st order thermal expansion coeff.	Beam propagation coeff.
Transistor, (SPICE)	Physical Model, Experimental data fitting	Real-ray tracing, Physical optics modeling, Optimization, toleranceing	Finite Element analysis	Auto-Cad

A simple example system using these boxes is shown in Figure 2. This shows inputs and outputs of the system as digital electronic signals. Also shown are the electronic power signals for the electronic and optoelectronic units. A laser system is shown as a power source for the optoelectronic conversion module, which in this case would imply a modulation technique rather than the use of direct sources. The generic OIPU (optical interconnection and processing unit) would be further characterized in expanded figures, representing lower levels of abstraction and decomposition.

Models

Tied to these basic blocks, we need three kinds of models *functional models, physical models,* and *parametric models*. Functional models are similar to those typical in digital electronic CAD. They allow us to answer the question: What does the system (as designed) do? We can use them for simulation and analysis of the behavior of the system operating on its input values and generating output values. The models themselves can exist at different levels of abstraction. Examples of high-level functional models are behavioral models. For instance, the equation:

$$\vec{F}(x, y, t_i) = \vec{a}(x, y, t_{i-k}) \wedge \vec{b}(x, y, t_{i-l})$$

states that the function of a block is to compute the logical *and* of two binary arrays that have values separated in space, and therefore in time. Other types of functional models are geometric optical models, as shown in Figure 3, and discussed later, and optoelectronic device models, analogous to spice models for electronic devices.

Physical models answer the question: What does the system look like? We need three dimensional physical models for the optical paths in the system, as well as more traditional "optical cad" for example, lens design.

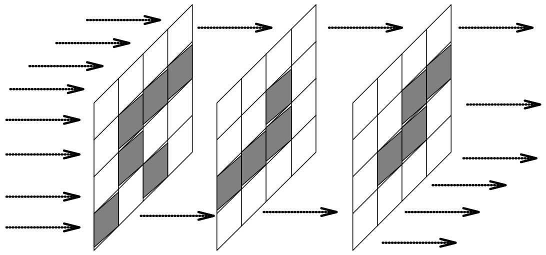


Figure 3. Functional Optical Model (OSLM)

The third types of models we need are parametric models. We need these models to answer the question: Can we build a system with the required specifications using the components we have chosen? For electronics, these would be parameters like the bandwidth, and power consumption of the components. Similarly, in optics we need to consider all the characteristics of the components that affect the quality of the information which flows through the system.

At each abstraction level which we choose to model, we need to define the input/output characteristics of the basic blocks in terms of "characteristic parameters". As examples, the characteristics of some optical components are shown in Table 2. Not all of these parameters are appropriate for all abstraction levels.

Also, in order to perform analysis, we need to define a corresponding set of characteristic parameters of the electronic and optical signals in these systems. In general, theses parameters should be appropriate for engineering or design decisions, rather than at the level of fundamental physics. Figure 4, based on [7], shows the full range of abstraction which could be used for modeling optical signals propagation. The most basic model is called Ray Optics, or geometrical optics where we use simple geometry and the normal of the propagating wave. Gaussian Beams are models of paraxial waves, a simplification of more general Wave Optics, which uses scalar waves to model propagation. More general is Electromagnetic Optics where the true E/M Fields are directly modeled. Finally, Quantum Optics or quantum electrodynamics are

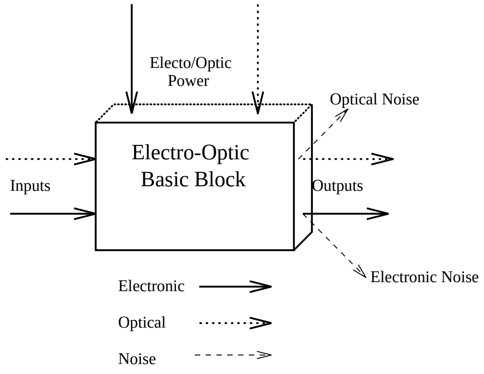


Figure 1. Basic Block

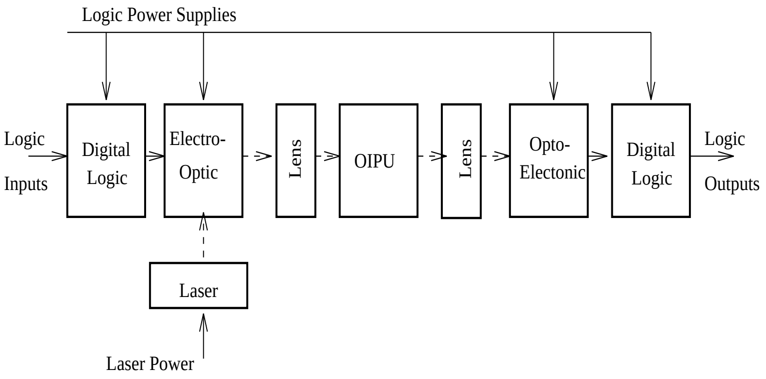


Figure 2. FSOI System

Table 2. Some Devices and Properties

Device	Properties
Lens	material, λ, polarization, losses, reflection, MTF, PSF, phase map, geometry, aberration, absorption
Polarizing beam splitter	size, λ, contrast, S/P, angle, polarization
Spot array generator	λ, number of spots, distance, spot size, spot uniformity, spot spacing, geometry
Optical isolator	intensity out, intensity in, absorption λ
Beam collimator	size &shape of input beam, λ, size & shape of output beam, polarization
Laser	power out, power in, λ, δλ, modes, duty cycle, CW, modulation, size shape, solid angle, polarization

required to model propagation in certain materials. As shown below Ray optics are appropriate for the most basic models, while Gaussian beam optics might be more appropriate for typical FSOI applications.

Based on the characteristic parameters of the signals that carry information between the components, we can then define the parametric models of the components in terms of the ways they *transform* the characteristic parameters of the signals. A partial list of the parameters of optical and electronic signals is shown in Table 3. It is important to note that optical signals are inherently based on modulation of a “carrier” of light. The characteristics of this carrier are as important to model as the signal itself. Analogous to the way we to model components, we need have a flexible model for the *propagation* of optical signals as well.

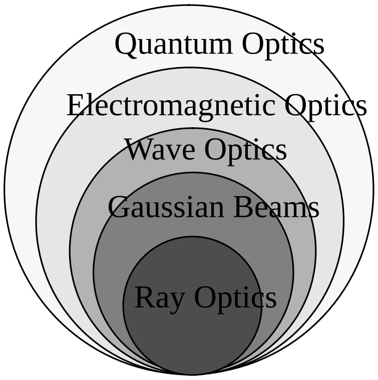


Figure 4. Models of Light

Table 3. Some Attributes of Electronic and Optical Signals

Electronic Signals	Optical Signals
amplitude-phase	amplitude-phase
signal spectrum	signal spectrum
noise spectrum	noise spectrum
modulation	modulation
pulse width/ spectrum	pulse width/ spectrum
power	power
	coherence
	light spectrum
	polarization
	spatial distribution
	spatial mode content

Parametric Analysis

Once we have the input/output characteristics of the basic blocks (at whatever abstraction level) we can begin to perform the analysis to determine the input/output characteristics of the entire system. This kind of analysis is more similar to current analog CAD tools which work in terms of frequency response, slew-rate, Q-factor, etc. [5] than the kinds of analysis typically done for digital electronic systems.

Even in systems as abstract as Figure 2, we can assign parametric specifications for the components and therefore begin to characterize the system behavior. As an example of the type of analysis that might be done, the left side of Figure 5shows the relationship of a signal waveform to the probability density functions, $p_m(x)$ of signal amplitude values for the logic levels zero and one, with the sum of all noise sources assumed to have a Gaussian distribution:

$$p(x) = \frac{e^{-(x-m)^2 / \left(2\sigma^2\right)}}{\sqrt{2\pi}}$$

Here m is the mean value for the logic level and σ is the standard deviation of the distribution. If we consider a receiver in the opto-electronic sub-system that has a single threshold value at $\frac{m_1 - m_0}{2}$, the shaded area below the curves shows the probability of a bit error. That is, the probability of either a zero being detected as a one or a one being detected as a zero. Following the derivation in [6], for this system the relationship of the probability of a single bit error occurring, P_e, to the signal to noise ratio (S/N) is:

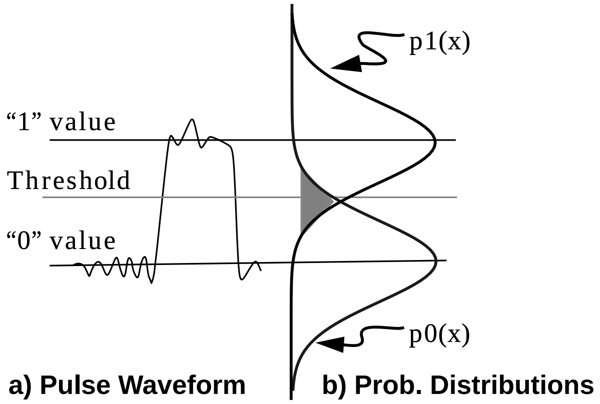
$$P_e = \frac{1}{2}\mathrm{erfc}\left(\frac{(S/N)^{1/2}}{2\sqrt{2}}\right)$$

where **erfc** is the complementary error function. This defines the bit error rate (BER) of the system and is plotted against the signal to noise ratio (in dB) in the right half of Figure 5.

We should note that real systems have many noise sources (including those in the receiver itself) that are often not Gaussian. Cross-talk between signal channels is also a noise source. On the other hand, modulation and coding methods, as well as differential signaling, and other techniques can be used to reduce the bit error rate. These would all have to be modeled in the analysis system.

Functional Analysis Examples

To provide a simulation framework for the models we are developing, we are investigating the use of Ptolemy [8] which is a generalized simulation framework used for rapid prototyping of digital signal processing systems. By using a pre-existing framework, which already has support for system software, digital simulation, and signal processing we hope to build an integrated system which meets the needs of optoelectronic systems design.



a) Pulse Waveform **b) Prob. Distributions**

Figure 5. Parametric Model (BER)

Figure 6 was generated by using extensions to Ptolemy to support simple free space optical systems. The top figure is used an input array of 128x128 light sources. The center figure shows a simulation of paraxial rays to model the functionality of a 2x2 lenslet array performing a fan-out operation. The lack of density in the output is a result of using single light beams from each source. This model gives only the most basic functional information about how the lenses work.

In order to address issues such as power and cross talk, the simulation shown at the bottom of Figure 6 moves to a

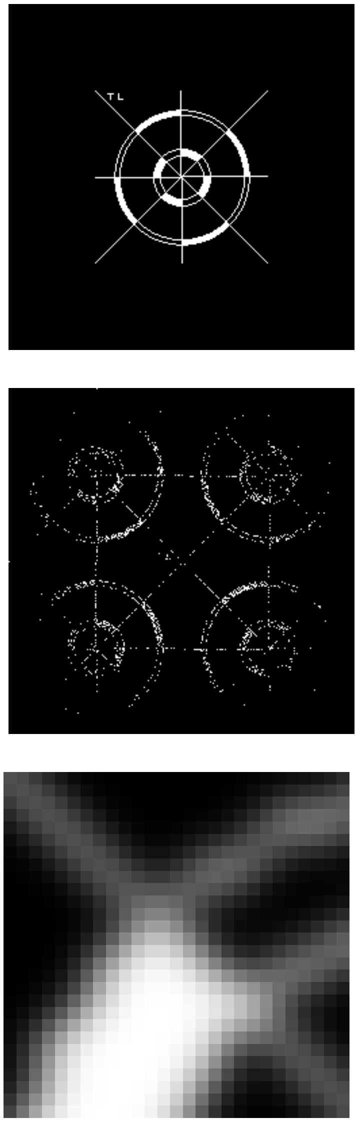


Figure 6. Optical Functional Simulation

lower level of abstraction using gaussian beams. We can now perform more detailed analysis. This figure shows a small piece of the image as seen by an array of detectors integrating the power from each beam in the source array.

A second example using Ptolemy is shown in Figure 7. here the figure depicts the use of Ptolemy components at the level of ray optics to simulate the optical symbolic substitution method presented by Brenner[9]. The input image Figure 7(a) is read in then split by an ideal beam splitter. The split images are then spatially shifted, with the amount and direction of shift determined by the search pattern. Next, the shifted images are superimposed upon each other, inverted and then masked with the mask pattern shown in Figure7(b); the mask pattern is also dependent upon the search pattern. Now that the search patterns are located, the image is re-split and spatially shifted according to a scribing pattern. Finally, the shifted images are superimposed to produce the output image shown in Figure 7(c).

Using Ptolemy has several advantages. In particular it provides a simple graphical user interface, a built in simulation engine, and a method for iteration. The real advantages we expect to realize with Ptolemy are its support for varied domains, simulation across those domains and access to existing signal processing libraries. Of course the disadvantage of using Ptolemy is that we are tied to another tool developer's code and have to live with their decisions.

Discussion

One important issue in the discussion of these tools is their usefulness in light of the wide variety of commercially available single-discipline CAD tools (i.e., for analog or digital electronic circuit design, for geometrical or physical optical system design, for FEA thermal analysis, etc.). An effective FSOI system should concentrate on two main issues: 1) Identification of required design areas or tasks for which no useful CAD tool exists 2) Creation of system level interfaces to provide simple, rapid and more intuitive generation of the inputs to, and filtering/analysis of the outputs from these other tools.

A key requirement of the design system will be the need to provide analysis tools at the optoelectronic systems level. At this level the design system must support parametric modeling and analysis of the sub-systems in order to allow the designer to perform design trade-offs in terms of both technology and architecture. This will improve the system design task in several important ways. It will enable a system architect to perform truly system-wide "what-if" analyses, rather than only analyzing a subset of the system constraints. Certain aspects of the system design are themselves very multidisciplinary. System and device level packaging is one example, and the lack of a widely accessible tool to facilitate this design task and quantify the trade-offs is becoming quite apparent in the FSOI community. A tool which provides a conceptual interface to powerful modeling and optimization programs will enable those programs to be used by researchers and designers will little experience in that aspect of the design space.

Since few system designers, or sub-system or component designers, for that matter, can afford to spend 100% of their time doing design, they often must re-learn parts of the design programs each time the use them, and they rarely fully utilize all of the program's capabilities. A CAD tool which accelerates this "re-acquaintance" would be invaluable. Additionally, many of these low-level programs produce prodigious amounts of data from even relatively simple analyses. Filtering the critical information from the pages of data produced is a skill developed by experienced operators, but automating this filtering process would make the full capabilities of these programs available to more system level designers.

Levitan, Chiarulli and Rempel would like to acknowledge the partial support of AFOSR Grant F-49620-93-1-0023 and NSF Grant MIP-9421777.

References

[1] A. McAulay. *Optical computer architectures: the application of optical concepts to next generation computers*. Wiley, New York, 1991.

[2] A. S. Daryoush, N. Samant, D. Rhodes, and D. Sturzbecher. Photonic cad for high speed fiber-optic links. *Microwave Journal*, 36(3):58, 61--2, 66--7, 69, March 1993.

[3] C. Bergstrom and J. Palais. Digital fiber optic network synthesis. *IEEE LTS*, 3(1):27--33, February 1992.

[4] M. de La Chapelle. Computer-aided analysis and design of microwave fiber-optic links. *Microwave Journal*, 32(9):179--86, September 1989.

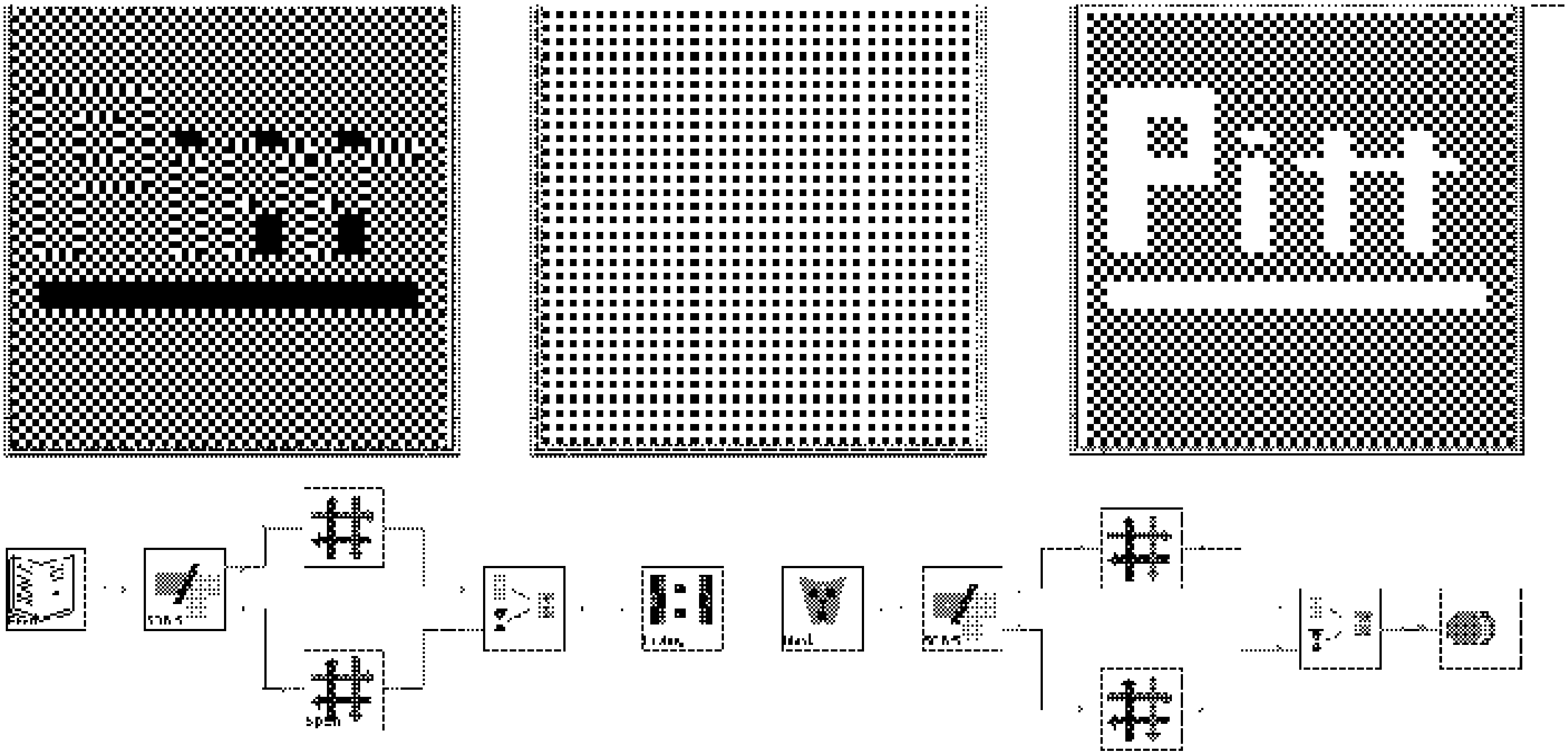
[5] R. Harjani, R. Rutenbar, and L. Carley. Oasys: a framework for analog circuit synthesis. *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, 8(12):1247--66, December 1989.

[6] J. M. Senior. *Optical Fiber Communications Principles and Practice*}. Prentice/Hall International, Englewood Cliffs, N.J., USA, 1985.

[7] Saleh, B.E.A. and M.C. Teich, *Fundamentals of Photonics.* Wiley, New York, 1991.

[8] J. Buck, S. Ha, E. A. Lee, and D. Messerschmitt, "Ptolemy: A Framework for Simulating and Prototyping Heterogeneous Systems," *Int. Journal of Computer Simulation,* special issue on "Simulation Software Development", January, 1994.

[9] Karl-Heinz Brenner, Alan Huang and Norbert Streibel, "Digital Optical Computing with Symbolic Substitution", Applied Optics, pp. 3054-3064, Sept. 1986.



(a) Input Image

(b) Mask

(c) Output Image

Figure 7.Optical Symbolic Substitution