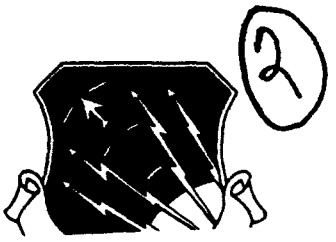
RL-TR-93-260
Final Technical Report
December 1993



AD-A278 129

DESIGN OF FREE SPACE INTERCONNECTED SIGNAL PROCESSOR

Rutgers University

Miles Murdocca and Thomas Stone

APPROVED FOR PUBLIC RELEASE; DISTRIBUTION UNLIMITED.

Rome Laboratory
Air Force Materiel Command
Griffiss Air Force Base, New York

This report has been reviewed by the Rome Laboratory Public Affairs Office (PA) and is releasable to the National Technical Information Service (NTIS). At NTIS it will be releasable to the general public, including foreign nations.

RL-TR-93-260 has been reviewed and is approved for publication.

APPROVED:

JAMES M. BATTIATO
Project Engineer

FOR THE COMMANDER

LUKE L. LUCAS, Colonel, USAF
Deputy Director
Surveillance & Photonics

If your address has changed or if you wish to be removed from the Rome Laboratory mailing list, or if the addressee is no longer employed by your organization, please notify RL (OCPB) Griffiss AFB NY 13441. This will assist us in maintaining a current mailing list.

Do not return copies of this report unless contractual obligations or notices on a specific document require that it be returned.

REPORT DOCUMENTATION PAGE

Form Approved
OMB No. 0704-0188

1. AGENCY USE ONLY (Leave Blank)	2. REPORT DATE	3. REPORT TYPE AND DATES COVERED
	December 1993	Final Apr 91 - Feb 93

4. TITLE AND SUBTITLE
DESIGN OF FREE SPACE INTERCONNECTED SIGNAL PROCESSOR

5. FUNDING NUMBERS
C - F30602-88-D-0028,
PE - 62702F Task 0051
PR - 4600
TA - P3
WU - PB

6. AUTHOR(S)
Miles Murdocca and Thomas Stone

7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES)
Rutgers University, Hill Center
Department of Computer Science
New Brunswick NJ 08903

8. PERFORMING ORGANIZATION REPORT NUMBER
N/A

9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES)
Rome Laboratory (OCPB)
25 Electronic Pky
Griffiss AFB NY 13441-4515

10. SPONSORING/MONITORING AGENCY REPORT NUMBER
RL-TR-93-260

11. SUPPLEMENTARY NOTES
Rome Laboratory Project Engineer: James M. Battiato/OCPB/(315) 330-7671
Prime Contractor: University of Dayton, 300 College Park, Dayton OH 45469-0001

12a. DISTRIBUTION/AVAILABILITY STATEMENT
Approved for public release; distribution unlimited.

12b. DISTRIBUTION CODE

13. ABSTRACT (Maximum 200 words)

Progress is described on a collaborative effort between the Photonics Center at Rome Laboratory (RL), Griffiss AFB and Rutgers University, through the RL Expert Science and Engineering (ES&E) program. The goal of the effort is to develop a prototype random access memory (RAM) that can be used in a signal processor for a computing model that consists of cascaded arrays of optical logic gates interconnected in free space with regular patterns.

The effort involved the optical and architectural development of a cascadable optical logic system in which microlaser pumped S-SEED devices serve as logic gates. At the completion of the contract, two gate-level layouts of the module were completed which were created in collaboration with RL personnel. The basic layout of the optical system has been developed, and key components have been tested. The delayed delivery of microlaser arrays precluded completion of the processor during the contract period, but preliminary testing was made possible through the use of other microlaser devices.

14. SUBJECT TERMS	15. NUMBER OF PAGES
Optical Computing, Photonic Switching, Birefringence, Optical Interconnects, SEED	48
	16. PRICE CODE

17. SECURITY CLASSIFICATION OF REPORT	18. SECURITY CLASSIFICATION OF THIS PAGE	19. SECURITY CLASSIFICATION OF ABSTRACT	20. LIMITATION OF ABSTRACT
UNCLASSIFIED	UNCLASSIFIED	UNCLASSIFIED	UL

TABLE OF CONTENTS

1. SUMMARY

This final report for Task P-0-6021 on contract F30602-88-D-0028 describes progress made on a collaborative effort between the Photonics Center at Rome Laboratory (RL) / Griffiss AFB and Rutgers University, through the RL Expert Science and Engineering (ES&E) program. The goal of the effort is to design and construct an all-optical digital processor making use of S-SEED [1] optical logic devices and free-space optical interconnects.

The effort involved the optical and architectural development of a cascadable optical logic system in which microlaser [2] pumped S-SEED devices serve as logic gates. The greatest portion of the effort was involved in the planning and preparation of experiments, and on the overall system design. Section 2 describes the goals of the effort, in the context of previous work undertaken at AT&T. A number of architectural issues were explored, which are detailed in Sections 3 and 4. At the time of the completion of the contract, the system had not yet been demonstrated because the microlaser devices had not been received. The system has since been demonstrated. Substitute microlasers were used in order to perform preliminary experiments, and the resulting analyses are detailed in Section 5. Section 6 describes a potentially significant application for this style of processor in the Air Force. The application involves reconfiguring a memory, as in swapping rows in a system of linear equations for a phased array radar system. Finally, Section 7 details special software that was used during the course of the effort.

2. INTRODUCTION

2.1 The AT&T System

The effort reported here is part of a larger ongoing project that was initiated in the Photonics Center in 1989. The initial goal was to develop a system that is similar in form to the S-SEED processor developed at AT&T Bell Labs in Holmdel, New Jersey [3], which is illustrated in Figure 1. In this model, four cascaded S-SEED arrays are interconnected with regular interconnection patterns in free

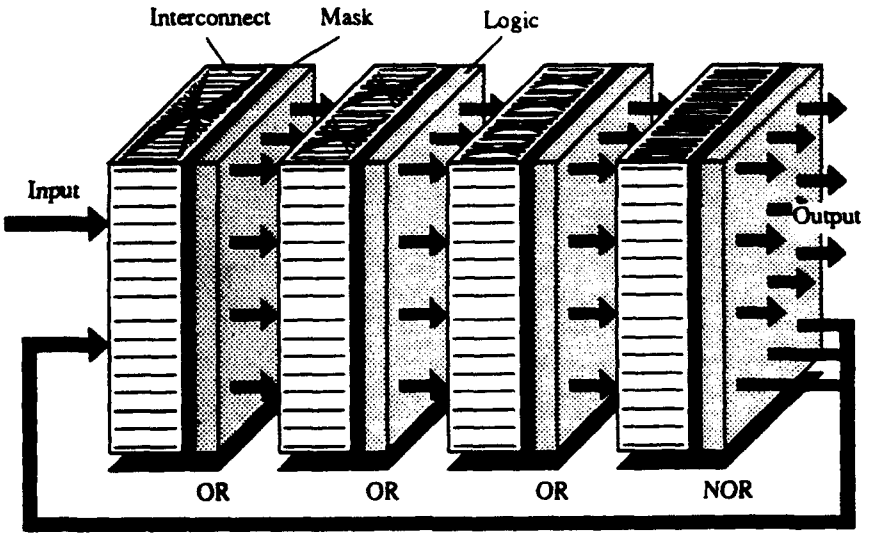
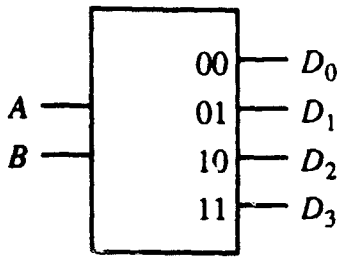


Figure 1: *Schematic of the AT&T Holmdel S-SEED processor.*



A	B	D_0	D_1	D_2	D_3
0	0	1	0	0	0
0	1	0	1	0	0
1	0	0	0	1	0
1	1	0	0	0	1

$$D_0 = \overline{A}\,\overline{B} \qquad D_1 = \overline{A}\,B \qquad D_2 = A\,\overline{B} \qquad D_3 = A\,B$$

Figure 2: *Block diagram and truth table for a 2-to-4 decoder.*

space. Fixed masks customize the system for specific functions. The AT&T system implemented a small programmable logic array (PLA). The actual demonstrated function was a 2-to-4 decoder, which is illustrated in Figure 2. The decoder translates a logical encoding on the A and B input lines into a spatial encoding on the D_i lines in which a different D_i is high for each AB pattern.

Figure 3 illustrates a circuit diagram of the demonstrated AT&T 2-to-4 decoder. A functionally equivalent electronic circuit diagram for a 2-to-4 decoder is shown in the figure. In comparison, the gate count for the S-SEED decoder appears exceptionally high for such a simple four-gate electronic function, which is a result of the experimental setup and is not fundamental to the methods [3]. For example, every signal must travel through a NOR gate on every level, which means that a relative inversion is not possible since every signal will go through the same number of inverting logic gates relative to every other signal. Thus, dual-rail logic is necessary, which increases gate count by approximately a factor of 2.

NOR logic is used at every level, and NOR is a nonassociative function, which means that in order to logically NOR three signals, two signals are first NORed, and this step is followed by an inversion, which is followed by a NOR of the result with the third signal. This method of dealing with the nonassociativity of NOR increases circuit depth.

Fan-ins and fan-outs are limited to two, which translates to a higher gate count than would be needed for a TTL approach in which fan-ins and fan-outs of 10 are typical. Fast logic implemented in emitter coupled logic (ECL) technology or in gallium arsenide (GaAs) technology typically has small fan-ins and fan-outs, which results in higher gate counts than with a smaller but denser complementary metal oxide semiconducting (CMOS) implementation. Thus, the low fan-ins and fan-outs are not uncommon for high performance circuits.

An electro-optical input to the system is not available. Inputs are provided by blocking light at the inputs of the top stage of logic devices, which produces true complementary 0's at the outputs of the first stage, and then selective blocking is used between the first and second stages to achieve the desired input pattern. This method of providing inputs introduces a cost of two additional rows of logic that would otherwise not be needed if an electro-optic interface is used.

There is some cost introduced by the fact that all of the signals travel through a logic gate regardless of whether a logic gate is needed at that level. This property of the architecture equalizes delays

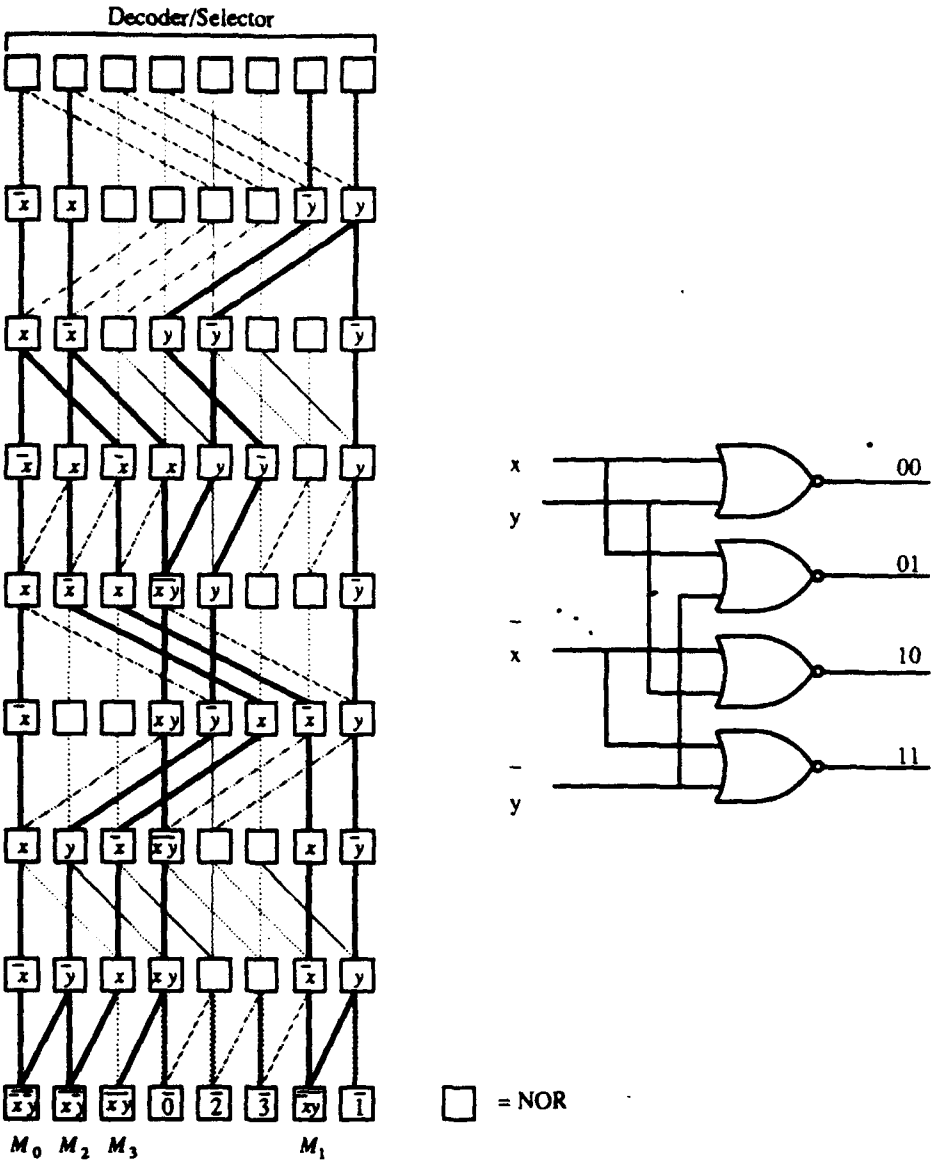


Figure 3: *The AT&T 2-to-4 decoder circuit (left) and an equivalent NOR circuit (right).*

between levels, similar to the way clock distribution is handled with tree structures in conventional digital electronics and can therefore be viewed as a benevolent cost. There is also a cost in forcing every level of logic to perform the same function, such as NOR instead of a mixture of AND, OR, or XOR (Exclusive-OR), which is normally allowed in more conventional electronic technologies. For a 2-to-4 decoder, this restriction does not affect the overall gate count, but for other applications it does.

Finally, the strictly regular interconnection pattern at the gate level introduces a significant cost in the gate count of the target machine. Although it has been shown by Murdocca *et al.* [4] that circuit depth and breadth are comparable to conventional electronic approaches for this model, the overall gate count is typically a factor of 4-8 greater than conventional electronics because of the forced

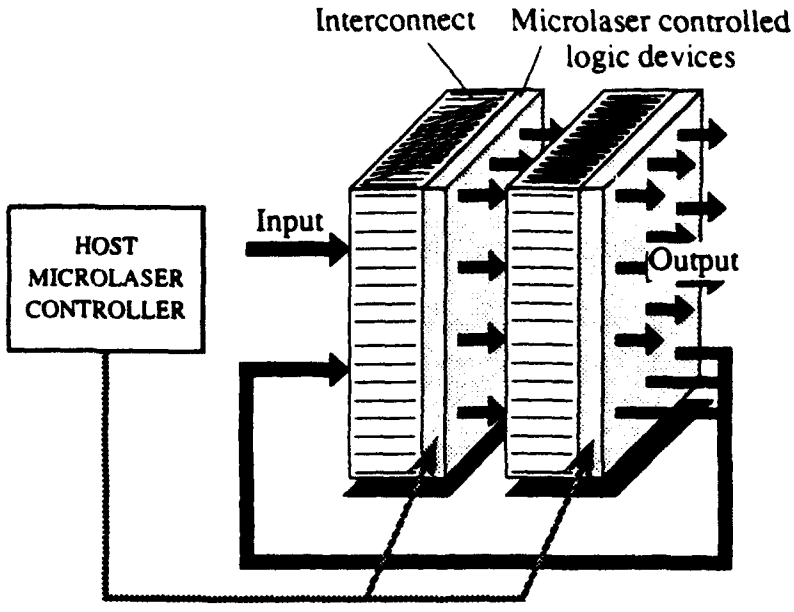


Figure 4: *Schematic of the Photonics Center S-SEED processor.*

regularity. This increase in gate count is balanced somewhat by greater utilization of the logic through gate-level pipelining.

2.2 The Photonics Center System

As the Photonics Center project matured from its inception in 1989, the device technology improved, and we focused on creating a simpler, but more flexible system than was demonstrated at AT&T. Unlike the four-stage AT&T system, the Photonics Center processor can use either two S-SEED arrays as illustrated in Figure 4, or can use a single S-SEED stage. Individual microlasers control each S-SEED mesa, which gives much finer control over the functions of the logic devices and provides greater optical power than was available in the AT&T system.

The model shown in Figure 4 consists of two arrays of optical logic gates and two stages of split-and-shift interconnects. The optical logic arrays are controlled by an electronic function generator via mating microlaser arrays. The microlasers perform the functions of the fixed masks. This is a significant improvement over the AT&T model, because it allows the masks to be reconfigured dynamically by selectively disabling microlasers through a host controller. Although the setup time for disabling a microlaser is limited by the speed of the electronic host controller (an HP 16500 logic analyzer/function generator for the Photonics Center processor), reconfiguration is a relatively infrequent operation for many applications, and so the relatively slow setup time is not necessarily a critical factor for the success of this type of system.

The target application for the overall project is an all-optical signal processor. The bulk of the effort reported here focused on the design of a small optical random access memory (RAM) that would serve as an integral part of a signal processor. Design and architectural issues relating to this application are described in the next two sections.

3. OPTICAL DIGITAL PROCESSOR

3.1. The Architecture

A conventional RAM consists of an address decoder and a means for storing bits. For our optical RAM, we designed the decoder to perform the same function as in a conventional RAM, but the stored bits are modulated by the microlasers. This variation is used only so that data can be input to the system by an electronic machine (an HP logic analyzer/function generator for this case). This would be replaced by an optical input mechanism in a finished system.

A number of RAM designs were developed for this model. Figure 5 shows one design, in which a fan-in of two and a fan-out of two are used for each S-SEED logic device. In order to disable an interconnection path, the source microlaser for that path is disabled. This disables the second output of the microlaser as a side effect. Thus, every logic gate has either two outputs or no outputs. Some logic gates in Figure 5 appear to have a single output, because those gates have a second output that is imaged off of the array and is therefore not shown.

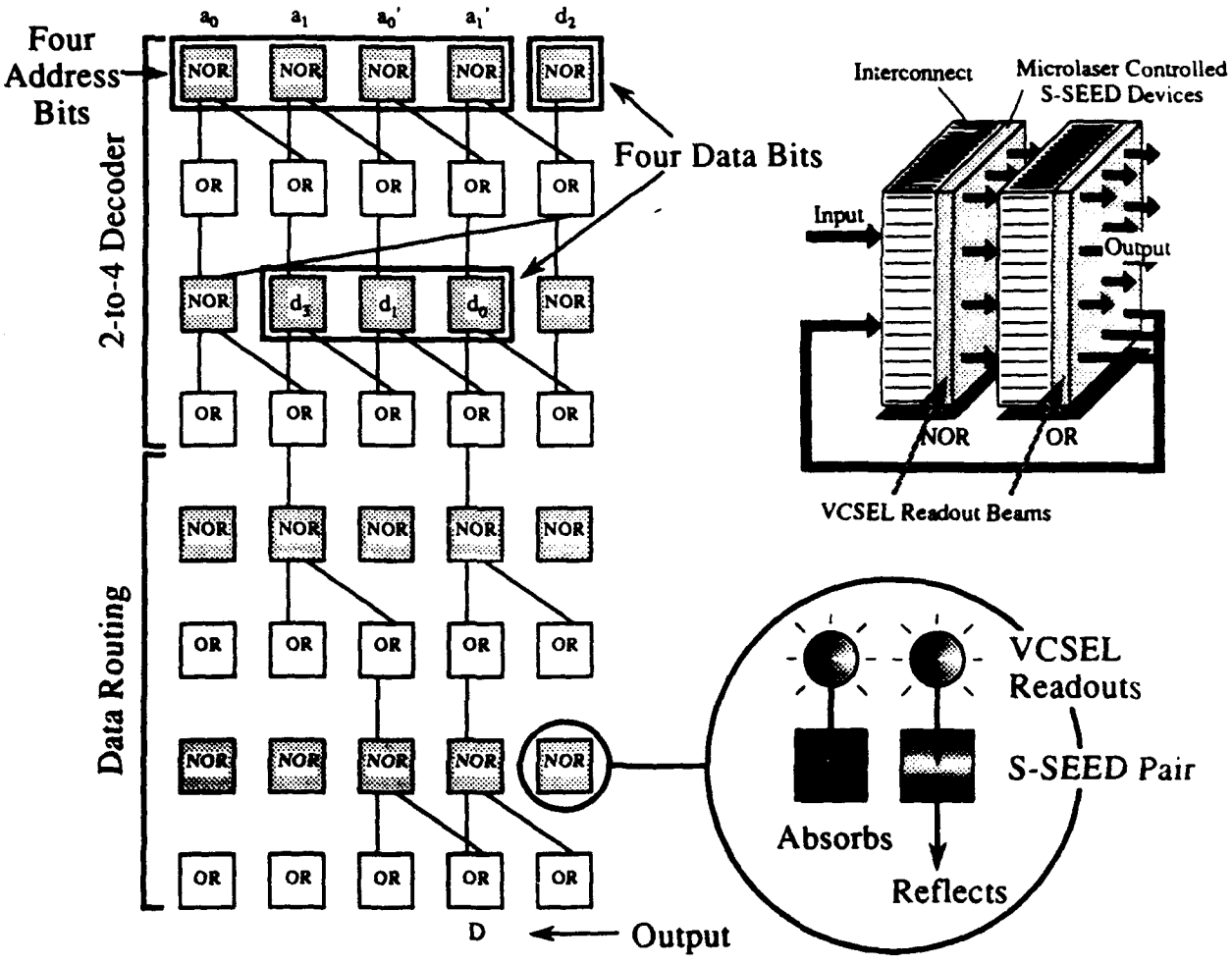


Figure 5: *Design of a 4 × 1 RAM for the Photonics Center S-SEED processor.*

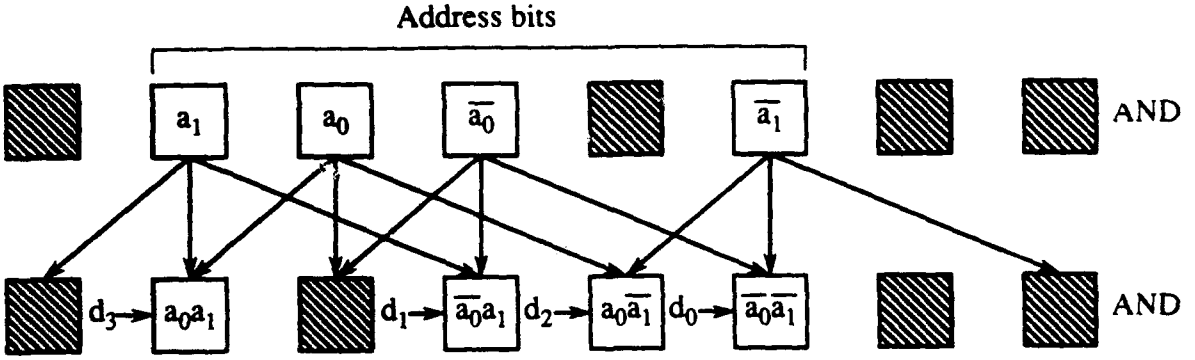
The circuit shows a 4-word × 1-bit optical RAM. There are two S-SEED arrays in the system: one for NOR, and one for OR. The interconnect from the NOR array to the OR array is a split-and-shift to the right by 1. The interconnect from the OR array to the NOR array is a split-and-shift to the left by 4. In order to pass the outputs of one row to the inputs of the next, the source row S-SEED windows must be illuminated. We need to block some of these beams in order to customize the circuit for a specific function, such as an address decoder for this case. In order to disable the output of a logic gate, we can place a mask in the image plane in a static approach as in the AT&T system. In our approach, the outputs are disabled by selectively disabling microlasers.

The four data bits d_i that are stored in the RAM are modulated by the microlasers that power the logic gates in the positions shown in the diagram. The address bits a_i are also modulated by microlasers. The one-bit output D is at the bottom of the diagram. The entire circuit fits into a rectangle that is five logic gates wide by eight logic gates deep, which gives an area complexity of 5×8 = 40.

3.2 Fan-outs and Fan-ins Greater Than Two

Figure 6 shows an alternative RAM design in which a fan-in of two and a fan-out of three are used. The circuit depth is reduced to two levels, and the gate count is reduced to 10 (five logic gates wide by two levels deep). As for the previous case, each logic gate has three outputs or no outputs, since interconnections are disabled at the source, which affects the fanned-out beams as well. In terms of area complexity, the fan-out of three approach is better. A fan-in of three is also possible, as well as greater fan-ins and fan-outs. As the fan-ins and fan-outs increase, however, the tolerancing requirements on the devices also increase. Since the microlaser devices did not arrive before the contract period ended, we were not able to measure device characteristics and settle on the degrees of fan-in and fan-out that could be supported.

A significant design problem was encountered in using fan-ins greater than two. The S-SEED devices consist of two mesas. If the relative intensity of light that is imaged onto one mesa exceeds the intensity on the electrically coupled mesa, then the device switches such that the mesa with the greater intensity absorbs incoming light. During operation, a preset cycle switches the S-SEEDs into a known state, followed by a data cycle in which the devices may switch back to the opposite state if the relative intensities of the incoming beams differ in the opposite way, and finally, a readout phase



(Data bits are modulated by microlasers, as in original design)

Figure 6: *A 4 × 1 RAM using a fan-out of three.*

allows the states of the S-SEED mesas to be read onto the succeeding stage (this is essentially another data cycle, which may need a preset cycle depending on how the system is configured).

If a fan-in of three is used, then complex results may occur. For example, if two high beams and one low beam are on one mesa and the complementary beams (two low and one high) are on the coupled mesa, what will happen? As we worked through the truth tables that describe three-input operation, we found that there was no way to apply presets such that the common logic functions AND, OR, NAND, or NOR could still be performed. The functions that we obtain are majority logic gates, which can be thought of as threshold logic gates.

In more detail, we assume that when the irradiance on one S-SEED window exceeds that on the other, the reflectivity of the more greatly irradiated window is switched to low, and if the irradiances are equal, then no change is made in the state of the device. Consider first a fan-in of three. There are three input spots imaged on each window of the S-SEED. The left side of Figure 7 shows a truth table for this case. With a fan-in of three the device acts as a "majority gate," in which the output depends only on the majority of the inputs. If two or more inputs are high, then the output is low, and *vice versa*. Thus the initial or preset value for the gate has no effect on the output. Another interesting case occurs with a fan-in of four, as illustrated in the right side of Figure 7. Here, the devices are again idealized and four spots are imaged on each S-SEED window. Since now there are cases in which equal numbers of inputs will be high and low (*e.g.*, where each window will have two "bright" spots and two "dim" spots), the preset state of the device is important as that state will then dominate. Although

A	*B*	*C*	*S*
0	0	0	1
0	0	1	1
0	1	0	1
0	1	1	0
1	0	0	1
1	0	1	0
1	1	0	0
1	1	1	0

Notes:
S shows the state of the *S* window in the R-S window pair.

P represents the preset state (either 0 or 1).

A	*B*	*C*	*D*	*S*
0	0	0	0	1
0	0	0	1	1
0	0	1	0	1
0	0	1	1	*P*
0	1	0	0	1
0	1	0	1	*P*
0	1	1	0	*P*
0	1	1	1	0
1	0	0	0	1
1	0	0	1	*P*
1	0	1	0	*P*
1	0	1	1	0
1	1	0	0	*P*
1	1	0	1	0
1	1	1	0	0
1	1	1	1	0

Figure 7: *A truth table for S-SEED operation using a fan-in of three (left) and a fan-in of four (right).*

we can use principles of threshold logic design here, we did not pursue this approach because it subdivided an already low contrast ratio.

Despite the relatively low contrast ratio, the use of greater fan-ins and fan-outs is still being considered for performing processing and memory functions. Practical constraints must be addressed, however, prior to using higher fan-in approaches with existing S-SEED devices. For example, lower contrast will be obtained for switching, and more spots need to be imaged onto the windows, *etc*. The first demonstration processor will thus be built using a fan-in of two while these other issues are addressed.

For the simple case in which a fan-in of two is used, an advantage of using our maskless approach is that the functions of the logic gates can be determined on-the-fly, based on the microlaser settings used in the preset cycle. For example, any combination of AND and NOR gates may be used on an array, or any combination of NAND and OR may be used, without modifying the physical interconnects. We did not take advantage of this capability in the RAM design because the regular structure of the memory only needed alternating stages of OR and NOR gates, but in a more general system, this capability may be significant.

3.3. Avalanche Mode Operation of S-SEEDs

We investigated two methods of maintaining a high throughput. In the first, which we call *delayed avalanche mode*, the clock (readout) pulses are applied to each successive row from a single signal, but this clock signal is passed through a delay loop from row to row. This approach permits the S-SEED devices to operate at their fastest natural speed, which is likely to be much faster than the speed through the control and logic drivers for the microlasers.

In a second approach, which we call *flash avalanche mode*, the devices are preset, and then all of the microlasers are turned on simultaneously. In effect all of the clock or readout beams are on at once. In this approach, each SEED window then has a high power input (readout) spot that reads its state, and two lower power input beams (logic beams) from the previous stage. All of these spots are present simultaneously. With an idealized device and system, the readout beams would be uniform in power and the difference in power between the high and low states of the logic beams would be used to switch the device. In practice, however, the large bias resulting from the simultaneous presence of the input and readout beams lowers the switching contrast which can limit the effectiveness of this approach with current devices. This is an asynchronous approach and care must be taken in the logic design to ensure that transient states do not alter the final output. For conventional combinational logic circuits this is not normally a problem, but the latching behavior of the S-SEEDs makes this an important consideration even for combinational logic circuits because race conditions may exist. We resolved that if we eventually replace the S-SEEDs with devices that have a better gain, such as HPT/microlaser CELL devices [5] (see Section 5), that an avalanche approach that does not suffer from race conditions may then be more practical.

A target clock rate on the order of 10-40 MHz is planned for the Photonics Center processor, but the effective clock rate is only 1/4 of that. The reason for this difference is that the S-SEEDs are operated on a four-phase cycle that consists of:

Phase 0: Preset S-SEEDs

Phase 1: Perform logic

Phase 2: Readout

Phase 3: Settle

As described above, the four phases are collapsed into a single phase for both avalanche modes, and so the maximum speed of the devices may be achieved for this approach. The idea is to perform presets on both arrays during an initial preset phase, then set up the data on the top row of the first array, and then let the signals propagate between the arrays in ping-pong fashion using a different row of devices on each pass. When the signals reach the bottom row, a global preset is applied to both arrays and the process repeats. The row-by-row operation is achieved by enabling a different row of microlasers on each pass for delayed avalanche mode, and simply by the natural maximum propagation speed between arrays for flash avalanche mode.

3.4. Optical Design Issues

Collaborative discussions were carried out between Rutgers and RL personnel that concentrated on design and construction issues. These included alternatives in the architecture and optical layout of the processor; the microlaser characteristics; mounting of the S-SEED and microlaser arrays; packaging and cooling issues with the microlasers; imaging the microlasers onto the S-SEEDs, the drive circuit/control requirements; and final equipment orders. Experiments were performed with a sample microlaser array (operating at 780 nm) in an effort to characterize microlaser behavior for use in the system. Although microlaser arrays operating at 850 nm were not delivered by the suppliers during the contract period, sample microlaser devices were used to try to understand and characterize uncertainties about the actual microlaser operating characteristics. For example, it is not clear if all elements will be linearly polarized along the same direction; how strongly the wavelength variation from laser to laser (and thermally induced) will be; how large the skews will be; how much power will be available; at what powers higher order spatial modes appear; *etc*. Activities prior to the closing date of the contract effort centered on preparing for the microlaser devices and identifying anticipated problems to the best extent possible in order to give the maximum flexibility in the memory/processor design. These activities with respect to microlaser issues are described in Section 5.

4. ARCHITECTURAL ISSUES

4.1. X-Y and Z-Folding

In mapping arbitrary digital circuits onto regular interconnects, such as the split-and-shift interconnect that we are using in the Photonics Center processor, we encounter problems in X-Y folding and in Z folding. The X-Y folding problem involves transforming a wide, two-dimensional circuit into a square three-dimensional (3D) structure that maps onto a cascade of optical logic arrays. The Z-folding problem involves folding a deep 3D circuit so that it fits in a shallow architecture. That is, when the number of logic stages in the circuit is greater than the number of logic arrays in the system, a deep circuit must be folded so that it maps onto the shallow physical system.

Under separate Rome Laboratory SBIR support, Rutgers student David Berger created X-Y folding algorithms for folding two-dimensional circuits onto square arrays using banyan and crossover interconnects. The split-and-shift interconnects are the first ones we will use in the S-SEED processor, and so we have also been looking at this form of interconnect for both X-Y and Z folding.

The following is a summary of Berger's findings:

- No more than three angles of connections are needed for each interconnection stage in a folded banyan circuit. This is important, because the folding process should not place greater demands on the optical interconnects than the original unfolded circuits.
- Any circuit can be trivially X-Y folded if it uses either a crossover or banyan interconnection network.
- The number of connections that change for each stage between an unfolded and a folded circuit is logarithmic in the width of the original circuit: lg(circuit width)/(size of fold). The significance of limiting the number of changes is that it determines the update time for some types of reconfigurable interconnects.
- Folding cascaded one-dimensional (1D) horizontal butterfly interconnects results in vertical butterfly interconnects.
- Folding 1D horizontal crossover interconnects results in either horizontal crossovers or vertical butterflies, depending on the period of the stage of the circuit being folded. The resulting folded interconnection pattern is more complex than the folded butterfly in the sense that more angles are needed per stage.
- For Z folding, removing r stages from a circuit that uses a banyan or a crossover creates 2^r groups of gates that can no longer communicate. In order to avoid this problem, a Z foldable circuit must use a perfect shuffle or a regular interconnect that is more like a split-and-shift than a banyan or a crossover.

From a designer's viewpoint, the banyan and crossover interconnects are easy to use, but the split-and-shift is more practical. Berger's folding procedures and observations for the banyan, crossover, and split-and-shift interconnects are described in the remainder of this section.

4.2. Folding Methods for the Banyan

A circuit of width N logic gates is referred to as an "N-Line." The logic gates are numbered 0–N-1 (from left to right) as shown in Figure 8. For a banyan interconnect, which is shown in Figure 8, let k be the size of a block that is used in the folding process. There are r blocks, of k gates each, where $r = n/k$ (n is the width of the unfolded circuit). Blocks are labeled from left to right, B_1–B_r. The operation B_iB_jC is defined as the i^{th} block being "C"ut away from the j^{th} block and placed on top of it, creating a new block B_{ij}. This action is shown in Figure 9 for an 8-line. $B_{ij}B_{lm}C$ places the block B_{ij} on top of block B_{lm}, forming a new block B_{ijlm} as shown in Figure 10.

For a given line width N, the block width k must be selected and a method must be found for folding the circuit. It is impractical to explore all $N!$ possible solutions, but the recursive layout of the banyan suggests that the solution for the simplest problem may be extended to larger problems. In our approach, a 4-line is folded, yielding six unique solutions. Of these, the solution arrived at by using

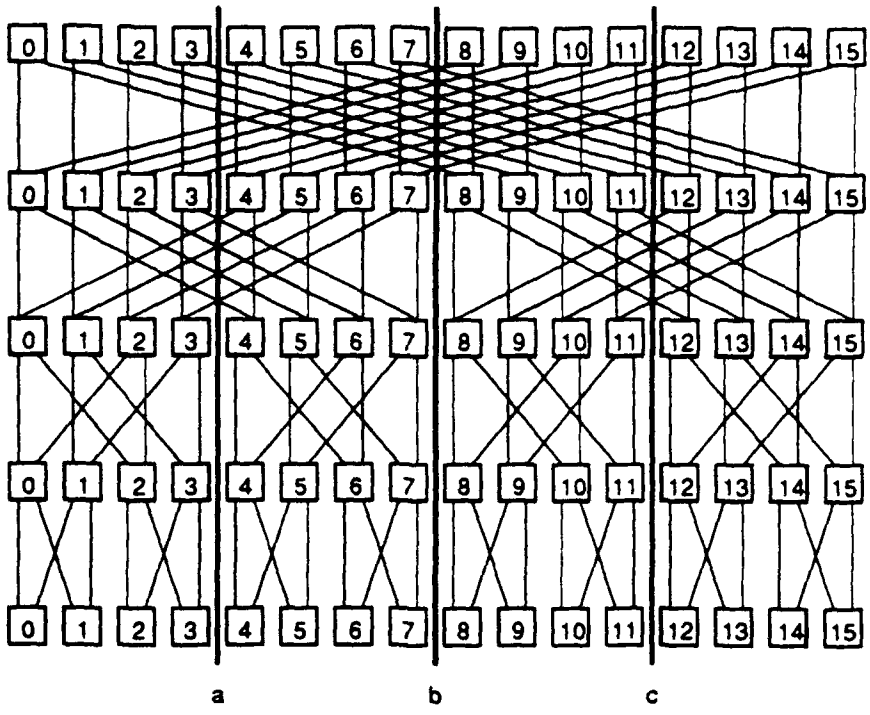


Figure 8: *Points a, b, and c indicate folds for a 16-line.*

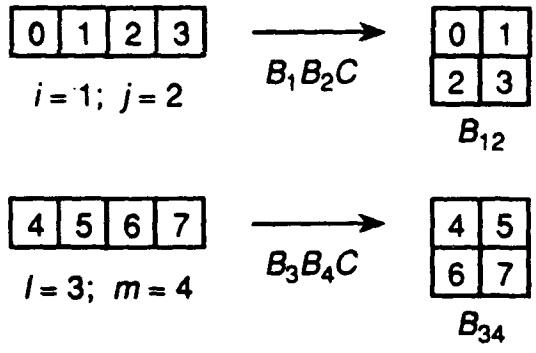


Figure 9: *An 8-line (0 – 7) is folded.*

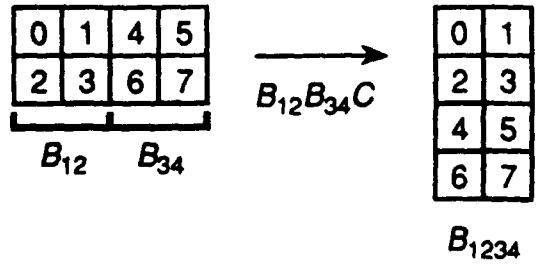


Figure 10: *An example of cutting.*

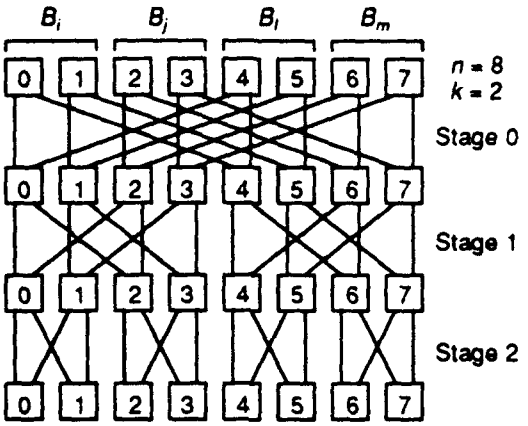
the cutting operation described above appears most appropriate. This solution is also simple to describe algorithmically for large circuits: First, take an unfolded circuit and choose k to be as large as possible. Then, as necessary, apply the cutting operation described above. For a 4-line that is mapped onto a 2×2 array, the 2D circuit is:

0 1 2 3

After folding via B_1B_2C, the 3D circuit is:

0 1

2 3



Operations:

B_1B_2C, $B_{12}B_3C$, $B_{123}B_4C$, $B_{1234}B_5C$, $B_{12345}B_6C$, $B_{123456}B_7C$, $B_{1234567}B_8$

Resulting Gate Layout:

0	1	2	3	D4	D4	D4	D4	D2	D2	D2	D2	D1	D1	D1	D1	R	R	L	L	R	L	R	L
4	5	6	7	D4	D4	D4	D4	D2	D2	D2	D2	U1	U1	U1	U1	R	R	L	L	R	L	R	L
8	9	10	11	D4	D4	D4	D4	U2	U2	U2	U2	D1	D1	D1	D1	R	R	L	L	R	L	R	L
12	13	14	15	D4	D4	D4	D4	U2	U2	U2	U2	U1	U1	U1	U1	R	R	L	L	R	L	R	L
16	17	18	19	U4	U4	U4	U4	D2	D2	D2	D2	D1	D1	D1	D1	R	R	L	L	R	L	R	L
20	21	22	23	U4	U4	U4	U4	D2	D2	D2	D2	U1	U1	U1	U1	R	R	L	L	R	L	R	L
24	25	26	27	U4	U4	U4	U4	U2	U2	U2	U2	D1	D1	D1	D1	R	R	L	L	R	L	R	L
28	29	30	31	U4	U4	U4	U4	U2	U2	U2	U2	U1	U1	U1	U1	R	R	L	L	R	L	R	L
Initial				Stage 0				Stage 1				Stage 2				Stage 3				Stage 4			

Figure 11: *A 32-line is folded onto an 8×4 array. The notation "Dx" and "Uy" means "down x positions" and "up y positions," respectively.*

This approach may be trivially extended to large N. Note that in general, the largest k is equal to the width of the target array.

The general folding algorithm for a banyan network is described by a series of operations B_1–$B_{i-1}B_iC$ with i varying from 2 to r. The resulting square or rectangle has blocks from top to bottom B_1–B_r. Figure 11 illustrates the folding of a 32-line into an 8 × 4 array.

The description of the new interconnection networks at the p^{th} stage of the banyan may be computed directly. Z stages of interconnects are changed where $Z = \log_2(n/k)$. Straight connections are maintained. Right and left connections become down and up connections respectively. At the p^{th} stage, there is a repeating pattern of $n/2^{p+1}k$ levels of straight and down connections followed by the same number of levels of straight and up connections. The distance that the up and down connections must travel at the p^{th} stage is $n/2^{p+1}k$. Figure 11 also shows the new interconnects for a 32-line.

4.3. Folding Methods for the Crossover

For the crossover, an N-line is decomposed into blocks that represent 4 gates each. There are thus r blocks where $r = n/4$, labeled from left to right B_1–B_r. The i^{th} block represents gates $(i-1) * 4$ through $((i-1) * 4) + 3$ as follows:

$$((i-1) * 4) + 1 \quad ((i-1) * 4) + 2$$

$$(i-1) * 4 \quad ((i-1) * 4) + 3$$

The formation of these blocks results in a reduction of the original circuit width by a factor of two as shown in Figure 12. Additional foldings of the width of the circuit are accomplished as described below.

B_iB_jRC is an operation where the i^{th} block is cut, rotated 180°, and catenated below the j^{th} block yielding a new block B_{j-i}. This notation represents block j atop block i with the minus sign in front of the i meaning that the block has been rotated 180° as shown in Figure 12.

The crossover, because it is more complex than the banyan in the sense that it is space variant as it is drawn on the page, requires a more complex solution. Taking the same approach as used with the banyan, a 4-line is solved. An extendable regularity that can be algorithmically described is found in only two of the six unique solutions. Of these, only one of the solutions maintains three degrees of freedom in larger circuits. While all of the interconnects from the circuit are altered, at each stage they remain horizontal crossovers or vertical banyans.

An algorithm that describes the folding of a crossover network addresses blocks in succeeding groups of four. From left to right, the following set of operations is performed on each group B_i, B_j, B_l, B_m: B_iB_jRC and B_mB_lRC. This set of operations results in a reduction of the width by a factor of two. The same operations are carried out on the new set of blocks to accomplish further reductions in width. Describing the new interconnection networks at each stage is somewhat complex. However, since there are no more than three degrees of freedom at any stage, empirical testing of the folded circuit determines the new interconnection network. To do this, at the p^{th} stage observe the number of the gate in the lower right corner of the completely folded circuit. Finding the destination gate for its output at the $p+1^{th}$ stage will immediately indicate the form of the new interconnection network.

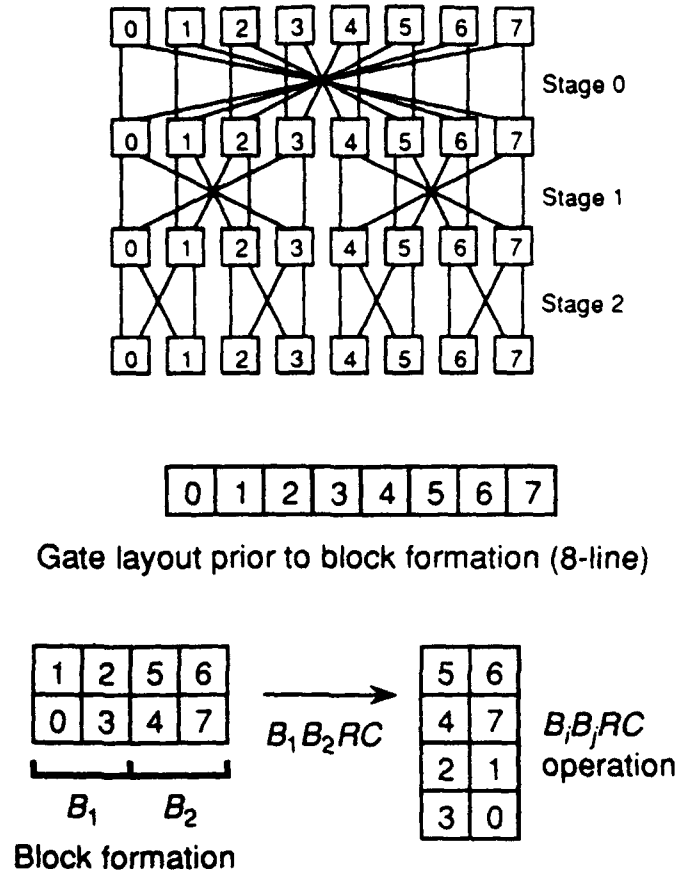


Figure 12: *Folding a crossover interconnect.*

4.4. Folding Methods for the Split-and-Shift

The large degree of variation in the forms of the split-and-shift interconnect make a simple folding algorithm difficult. However, a number of constraints can be placed on the design in order to simplify folding.

The split-and-shift interconnection network as described here is limited to two degrees of freedom at any stage. The connection angles that can appear at any one stage include straight and left, straight and right, right and left, left1 and left2, or right1 and right2. These pairs of shifts are illustrated in Figure 13. Since X-Y folding does not affect shifts that are straight, straight/left and straight/right combinations have special importance. With banyan and crossover networks, when an enabled (not masked) connection crosses a fold line, a different direction is needed to hit the appropriate gate at the next stage. This creates a new degree of freedom and is of prime concern with splits and shifts. The operation used to accomplish the folding is the same B_iB_jC used for the banyan, but here the size of the blocks are not as well defined. We divide the possible split-and-shift networks into two categories:

A) straight/left and straight/right

B) right/left, left1/left2, and right1/right2

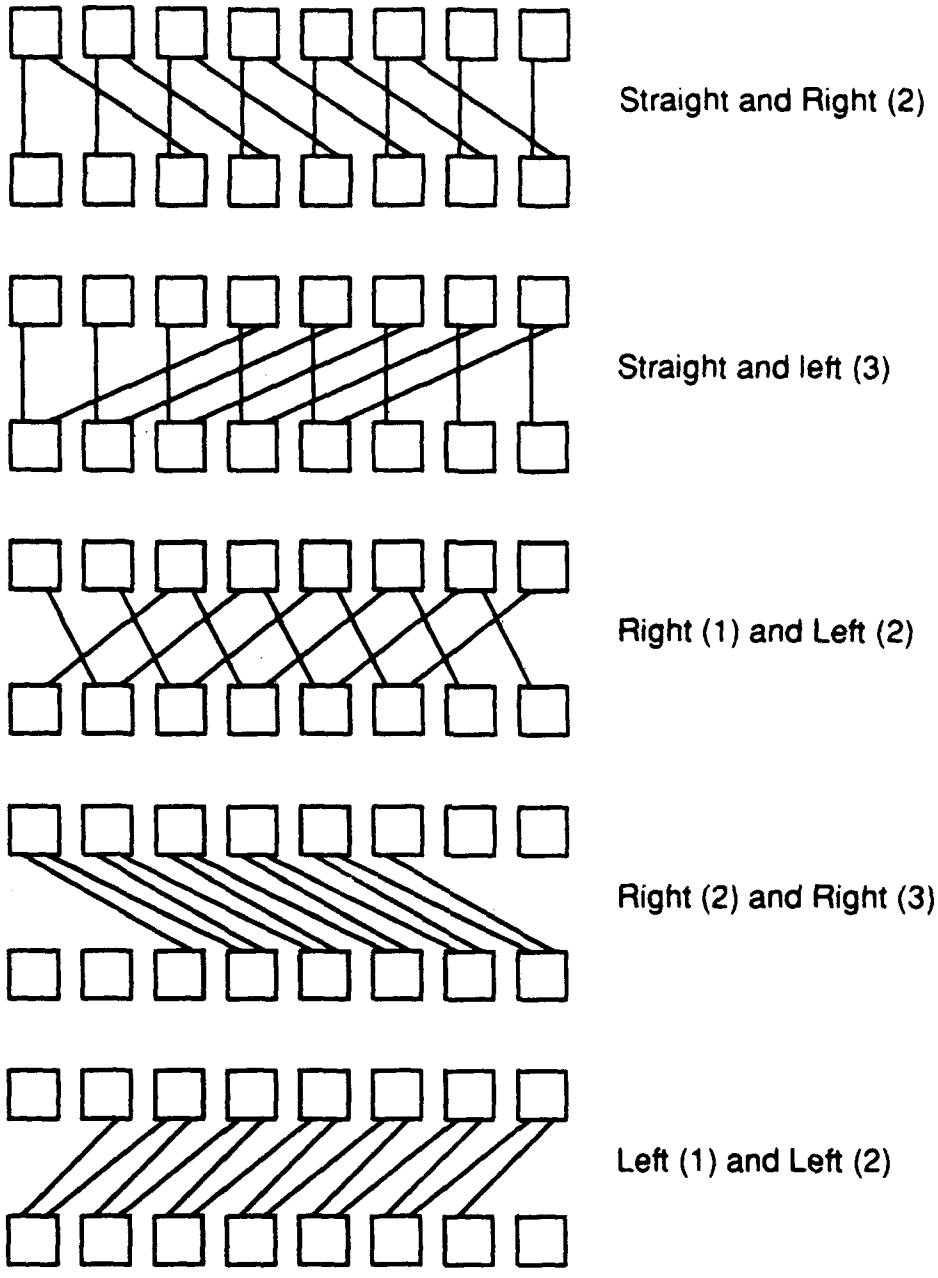


Figure 13: *A few topologies for a split-and-shift interconnect.*

When designing a circuit, following the rules described below guarantees that a split-and-shift circuit can be folded.

CATEGORY A

A designer needs *a priori* knowledge of where the folding points occur. Since straight connections are unaffected, the designer can use this knowledge to guide the use of shifts to the right or left. There are two approaches that permit the circuit to be folded. Either *no* shift can cross a folding point, or *all* shifts must cross a folding point. Recall that if the output of a gate crosses a folding point, a new

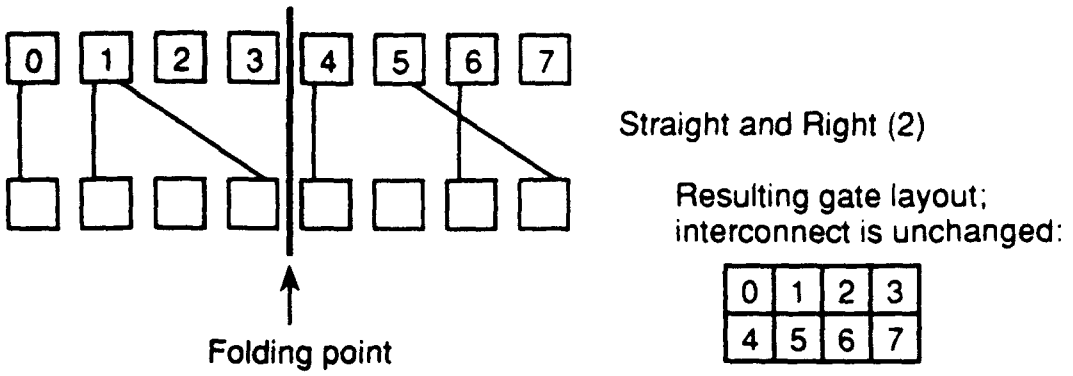


Figure 14: *No connections cross the fold line.*

degree of freedom is required, and our limit is two degrees of freedom per stage. Thus, if no shift crosses a folding point or the output of a gate whose shift does so is masked, as shown in Figure 14, the folded circuit has the same two degrees of freedom as before. If all of the shifts of a type cross the folding point, then they are all transformed into a different degree of freedom as shown in Figure 15, preserving two degrees of freedom. Note that the horizontal shift becomes a vertical shift in the folded circuit. The situation that must be avoided is when the output of some unmasked gates pass a folding point and others do not. Here the original two degrees of freedom remain, and an additional degree of freedom is introduced, which violates the constraints. This is illustrated in Figure 16.

CATEGORY B

A designer needs the same *a priori* knowledge as above. The fundamental difference between Categories A and B is that in Category B, there are no straight connections to ignore, thus two shifts must be considered. All connections that comprise the same horizontal shift (all right shifts, all left2 shifts, *etc.*) make up a group. There are two groups per stage. If all members of a group cross a folding point or none cross that point, then the circuit is foldable. The logic is the same as above. If no members cross that point, the same two degrees of freedom are maintained. If all cross that point, then they are all translated into a vertical shift. However, if only some outputs of a group cross that point, then an additional degree of freedom is introduced. Figure 17 illustrates both acceptable and unacceptable ways of folding this kind of network. From the figure, it is evident that it is difficult to

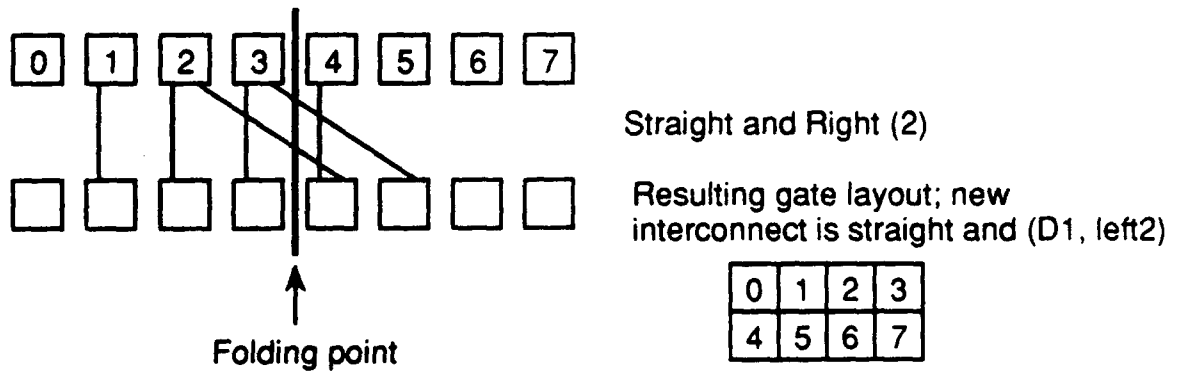


Figure 15: *All angled connections cross the fold line.*

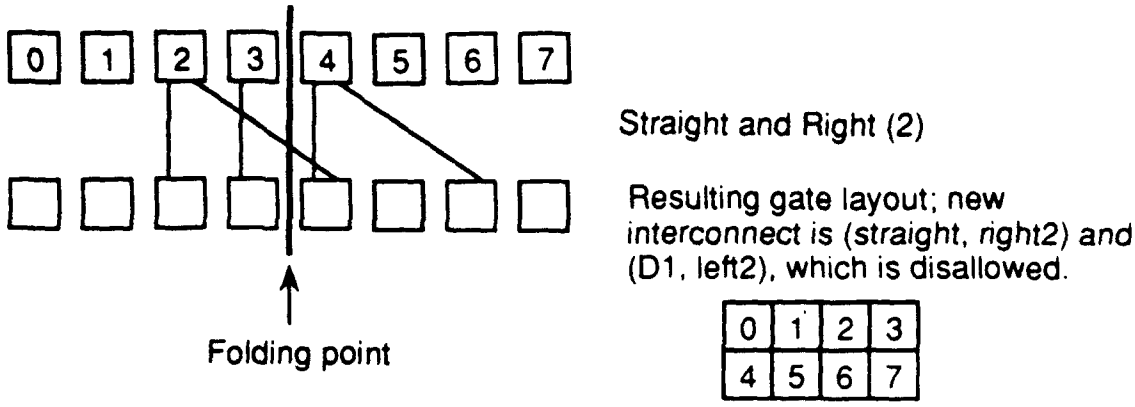


Figure 16: *A problem results when only some of the angled connections cross the fold line.*

have all outputs of a particular shift cross a point. For this reason, special attention should be given to the use of straight/right and straight/left interconnects.

4.5. Depth Mapping

Z folding deals with the problem of mapping circuit designs that are deeper than the physical system onto that system. Two problems that arise are caused by *depth mapping* and *relative inversions*. The depth mapping problem arises when there are more OR stages, using OR/NOR logic, in a circuit design than there are in the actual architecture. The relative inversion problem arises when the OR/NOR sequence of a circuit design does not match the physical system, and so the signals are sometimes incorrectly inverted.

Circuits that use OR/NOR logic are designed for a certain number of OR stages. Physical systems may or may not match this number of OR stages. Two scenarios arise in depth mapping. Let k be the number of OR stages in the design circuit and let l be the number of OR stages in the physical system. We have the following relations:

$k = l$ No mapping

$k < l$ trivial mapping

$k > l$ complex mapping

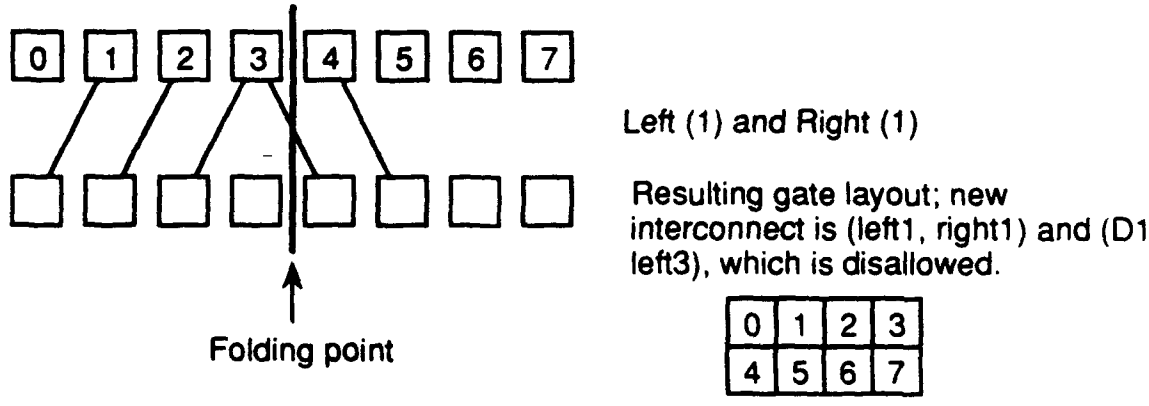


Figure 17: *A disallowed folding situation for two angled connections per logic gate.*

When $k = l$, the stages of the circuit design and the actual architecture match exactly. When $k < l$ there are excess OR stages in the actual architecture. The signals can be trivially passed through the remainder of the physical system using straight connections. In effect, the circuit design is Z-unfolded by being padded with l-k dummy OR stages. A problem arises when there are fewer OR stages in the actual architecture than in the circuit design. Banyan and crossover networks require a relaxation of optical constraints in order to be depth mapped Z-foldable. In illustrating this Z-folding problem, two assumptions are made. First, the physical system must have at least one OR stage, *e.g.* an OR/NOR system. Second, the interconnection network must be the same at every horizontal level within a stage, which follows our model for the Photonics Center processor. Based on these assumptions, it can be shown that banyan and crossovers are not depth mapped Z-foldable.

Removing r stages of interconnects from a banyan or crossover connected system creates $2r$ groups of gates that cannot communicate with each other regardless of the number of passes through the system. This means that it is impossible to achieve the connectivity of a removed interconnect with combinations of more or less granular interconnects. The perfect shuffle is identical from one stage to the next, and so the depth mapping problem is greatly simplified for this case. *Thus, we have found*

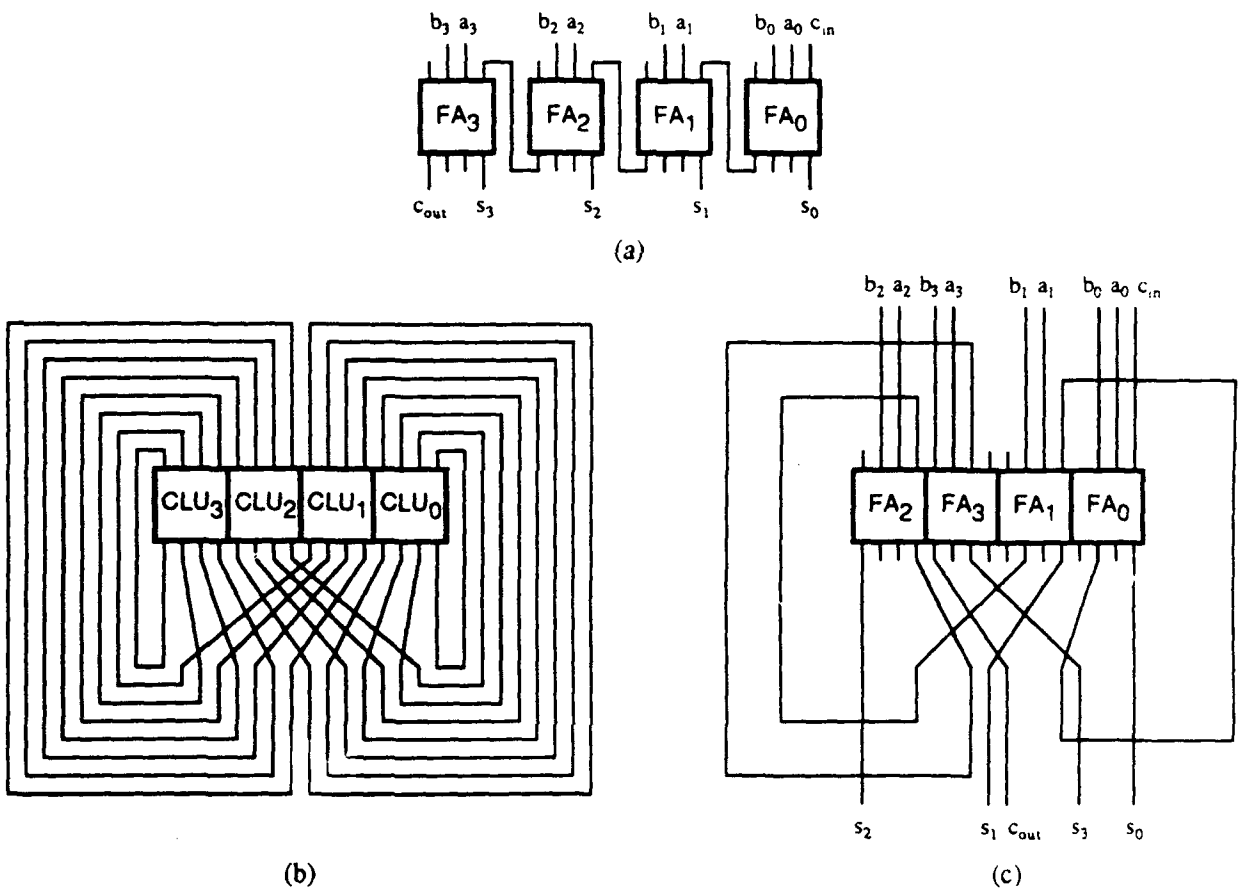


Figure 18: *The MSI interconnects of a 4-bit ripple-carry adder are mapped onto a single perfect shuffle. Original circuit (a); generic interconnection topology (b); and customized interconnect for the adder (c).*

that the perfect shuffle is one of the best regular interconnects to use for mapping purposes, while the split-and-shift is one of the best interconnects for implementation purposes.

4.6. MSI Mappings

For more complex circuits than a simple decoder, we can make use of the results of a related Rome Laboratory sponsored Phase II SBIR effort with which we are also involved. Our emphasis on designing circuits for the S-SEED processor has been at the gate level. We encounter special problems if we try to modularize our gate level components into standard medium scale complexity (MSI) components, while maintaining regular interconnects among the MSI components. Figure 18 illustrates a simple version of the problem, in which the MSI level interconnects of a conventional ripple-carry adder are forced into a single level of a perfect shuffle.

We know from permutation theory that a set of N inputs can be arbitrarily permuted at the outputs of a shuffle-exchange network of depth $3\log_2 N-1$. Here, $N=8$ and so the depth of such a permutation network is eight. The ripple-carry example is significant because it shows that the MSI interconnects for at least one circuit can be forced into a *single* stage of a perfect shuffle structure, rather than the theoretical upper bound, and so no depth is added to the circuit. We were concerned that this mapping might only have been possible because the example was so small, and so we doubled the width of

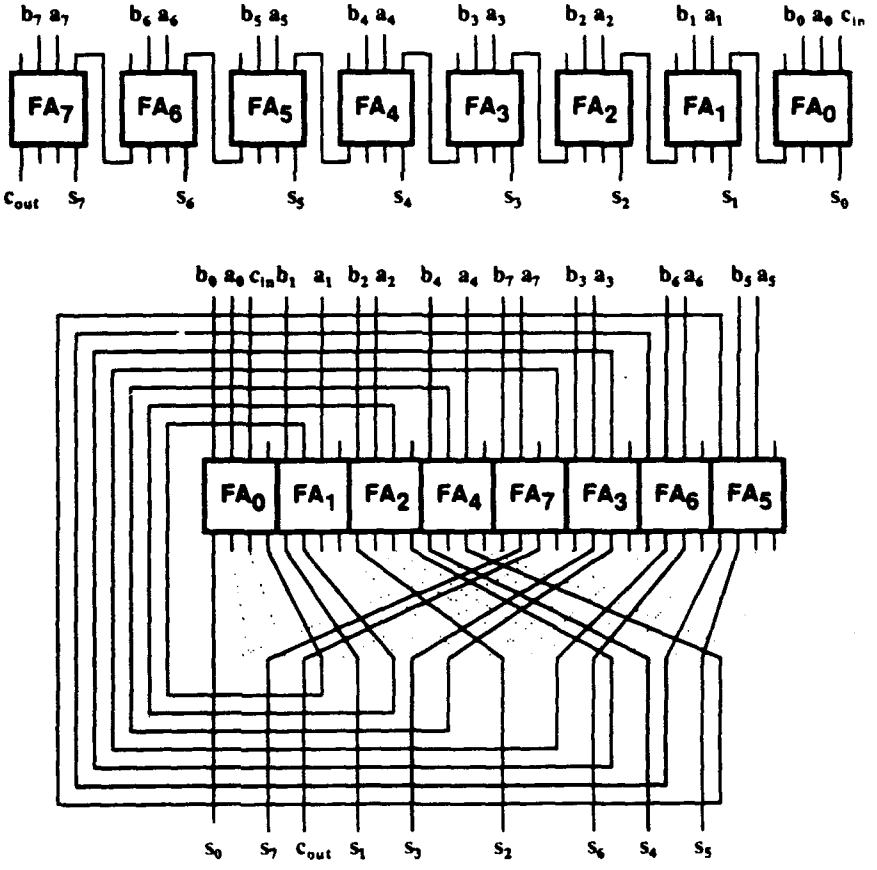


Figure 19: *An eight-bit ripple carry adder is mapped onto a single perfect shuffle.*

the adder, and found a mapping that still required only a single perfect shuffle stage. The solution for an eight-bit ripple-carry adder is shown in Figure 19. The abstract layout is shown at the top of the diagram, in which the FA_i are full adder modules. The perfect shuffle layout at the bottom of the diagram was obtained by feeding the connectivity pattern of the abstract layout into a program developed at Rutgers University (under joint AFOSR/ONR support – see Section 7).

A significant finding of the investigation is that regular structures map well onto a single stage of the perfect shuffle whereas irregular structures do not. Irregular structures are not impossible to map onto a single perfect shuffle stage, but they do require more work. Another case was investigated, which involves interconnecting PLAs for a 12-bit section carry lookahead (SCLA) adder. The schematic for this circuit is shown in Figure 20, which shows the layout as a digital electronic designer might draw it. Notice that the interconnects appear to be irregular, that the PLAs have different sizes as indicated by the varying numbers of inputs and outputs, and that there is fan-out within the interconnect itself. That is, a connection may be tapped in more than place. Since the point-to-point perfect shuffle does not support fan-out, the first step is to push the fan-out back to the originating PLAs. Figure 21 shows the remapped circuit. If an originating PLA is already at the maximum size allowed by the technology, then we must add a "fan-out" PLA. For this case, there is no need to add a fan-out PLA.

Again, we attempt to map all of the connections shown in this diagram into a single stage of a perfect shuffle. The problem is small enough to attempt an exhaustive search, using the permutation software described in Section 7, which failed for this case. There is no possible repositioning of PLAs with a 64-wide perfect shuffle that will work. An alternative approach involves repositioning the boxes (PLAs), and growing the smaller boxes up to the sizes of the larger boxes in order to absorb more input and output ports. This approach succeeded for the 12-bit SCLA as shown in Figure 22. Exhaustive search is no longer practical because the sizes of the PLAs vary, and so the best of the failed solutions for the original PLAs serves as a starting point, after remapping to remove fan-out, and then manual trial-and-error is used to obtain the solution.

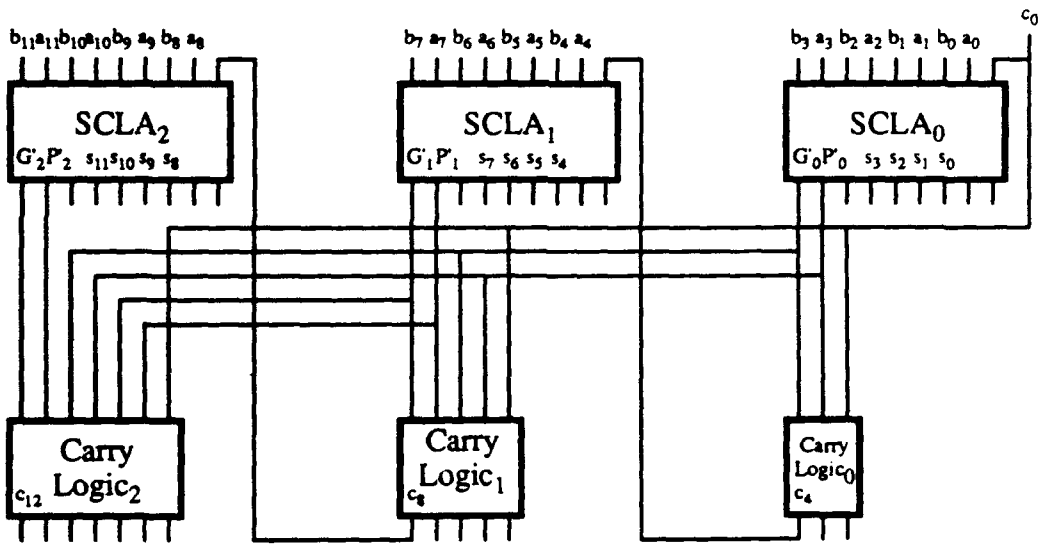


Figure 20: *Original SCLA circuit.*

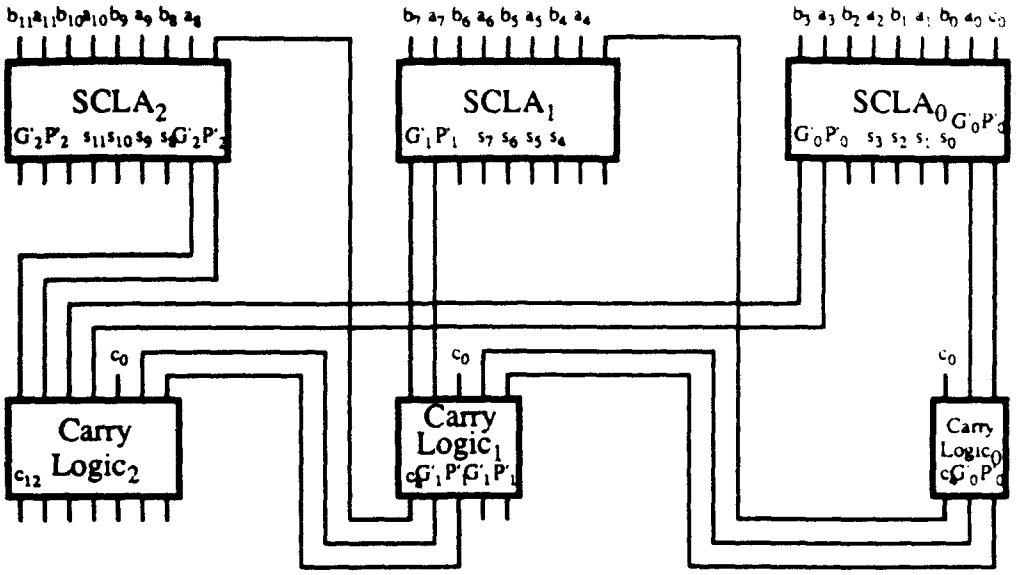


Figure 21: *Remapped SCLA circuit, with only point-to-point interconnects.*

A similar mapping was attempted for a 16-bit SCLA. The original schematic for the circuit is shown in the upper diagram of Figure 23. The first step is to push the fan-out from the interconnect to the PLAs. A problem is encountered, because the G_0 and P_0 outputs from $SCLA_0$ are fanned out to four other PLAs, and there are not enough unused output ports in $SCLA_0$ to produce four copies of G_0 and P_0. We can extend the width of $SCLA_0$ to create four more output ports, or we can add a PLA that fans out G_0 and P_0, and thereby avoid increasing the size of the largest PLA. The latter approach was attempted first. A fan-out PLA is added to the circuit, as shown in the lower diagram of Figure 23. The fan-out PLA produces two of the four needed copies of G_0 and P_0. The third copy goes to the Carry $Logic_0$ PLA, which produces the fourth copy of G_0 and P_0 at two of its unused output ports.

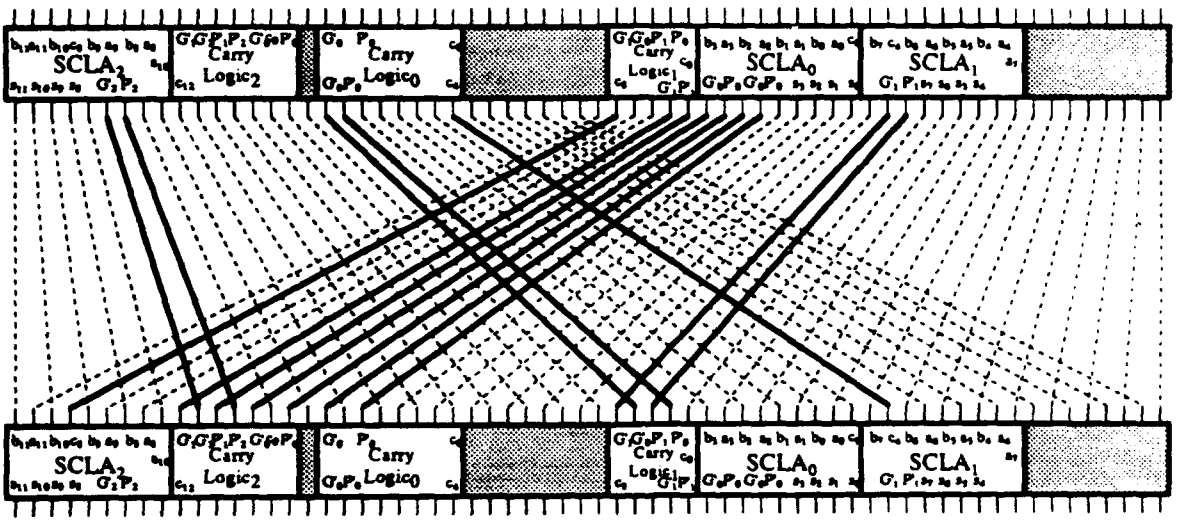


Figure 22: *A 12-bit SCLA circuit is mapped onto a single perfect shuffle stage.*

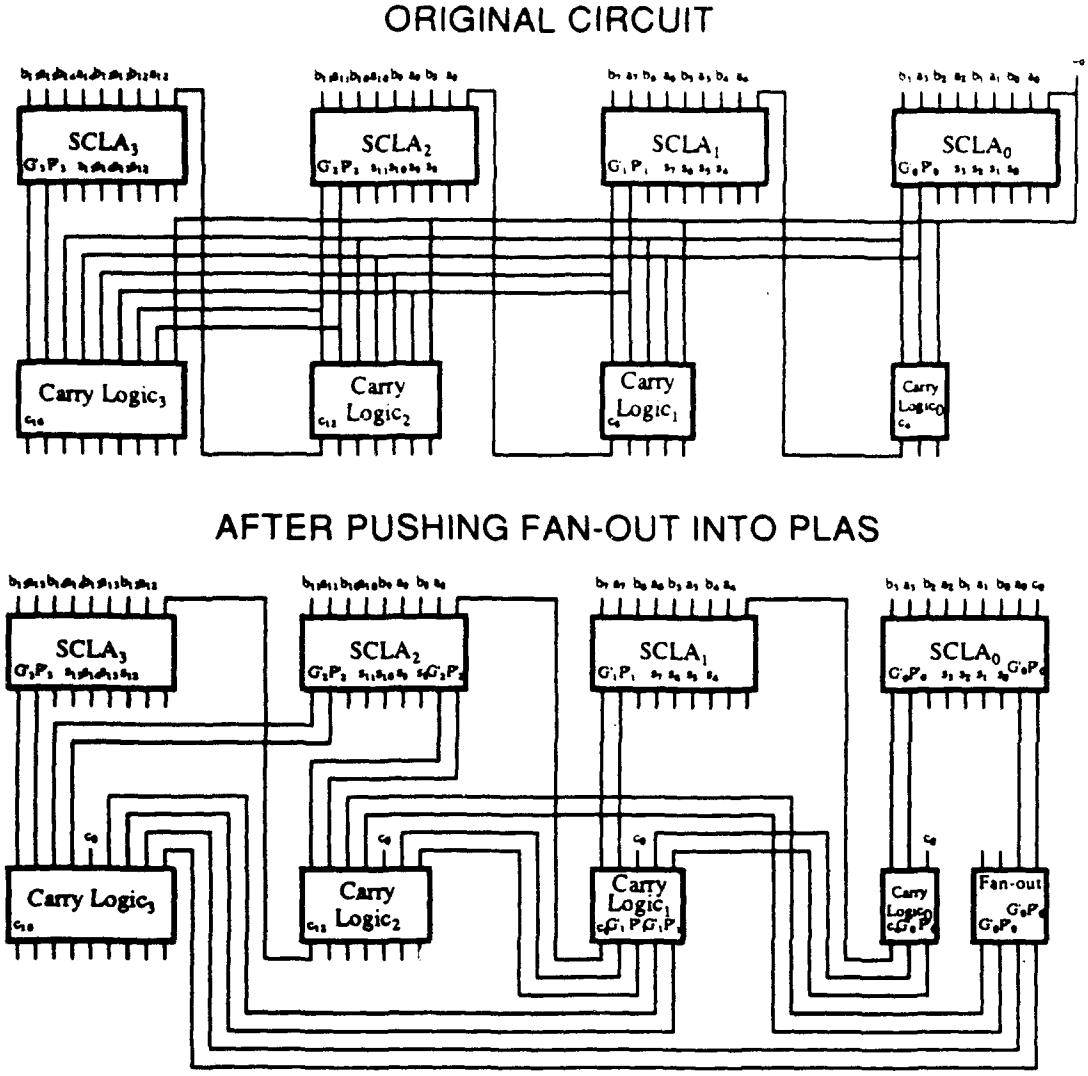


Figure 23: *Original 16-bit SCLA circuit (upper diagram) and remapped circuit with a fan-out PLA (lower diagram).*

As before, the best of the failed solutions for a 64-wide perfect shuffle serves as a starting point and then an attempt is made to manually grow the smaller PLAs up to the sizes of the larger PLAs. No solution was found using this approach, although a solution may exist. A solution was obtained, however, for the full 16-bit SCLA by extending the width of the $SCLA_0$ PLA, which allowed the fan-out PLA that was introduced in Figure 23 to be eliminated, and by extending the widths of some of the remaining PLAs. The width of the perfect shuffle was also doubled from 64 to 128. The solution is shown in Figure 24. We could not find a solution for the 16-bit SCLA using a single perfect shuffle stage that did not also increase the width of the widest PLA.

To summarize: a single point-to-point perfect shuffle interconnect without fan-out can implement all of the MSI-level interconnects. Although this is a significant accomplishment to achieve by hand, we have found no suitable method of fully automating the approach. At the moment, automatic MSI-

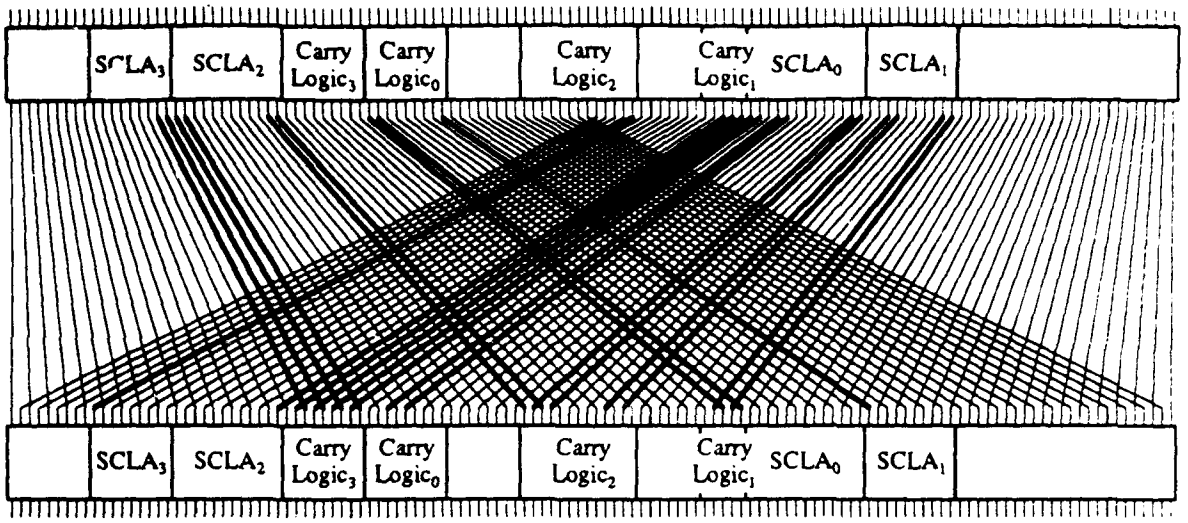


Figure 24: *A 16-bit SCLA circuit is mapped onto a single perfect shuffle stage.*

level interconnection using regular interconnects for irregularly structured circuits is an open problem.

For the S-SEED processor, we do not have the level of complexity in the physical hardware to demonstrate MSI interconnection, and so we did not make use of these findings in the reported effort. However, we did attempt a few circuit designs that maintained a strict split-and-shift interconnect at all levels, and came to the empirical conclusion that a perfect shuffle (or similar) interconnect is easier to work with at both the gate level and the MSI level. Thus, if we scale the S-SEED system to a larger size, the circuit design problem may be easier for a perfect shuffle type of interconnect than a split-and-shift interconnect.

5. SURFACE EMITTING MICROLASER ARRAYS

5.1 VCSEL Structure and Operation

A conventional diode laser found in a compact disk (CD) player is a few microns wide by several hundred microns long, and emits light parallel to the substrate. In contrast, arrays of vertical-cavity surface-emitting lasers (VCSELs) [2] that are only a few microns in diameter and about 6μm in height can be fabricated on ~10-100μm pitches (the center to center spacing). Depending on their construction, VCSELs can emit light at different wavelengths. Emission at 780 nm, 850 nm, and several other wavelengths has been demonstrated. Emission at 960 nm is especially interesting since a GaAs substrate is transparent at this wavelength, which allows detector/laser pairs to be monolithically integrated. We are working with S-SEED devices, and so we need VCSELs at a wavelength of 849 nm (+/– a few nm) in order to perform readouts. The S-SEED detectors accept a broader band of light, however, and so we actually use a wavelength of 856 nm for the preset operation.

VCSELs operate according to the same principles as ordinary semiconductor lasers. In a typical implementation, the amplifying portion of a VCSEL consists of a multiple quantum well (MQW) structure made up of epitaxially grown alternating layers of GaAs and AlGaAs. During operation,

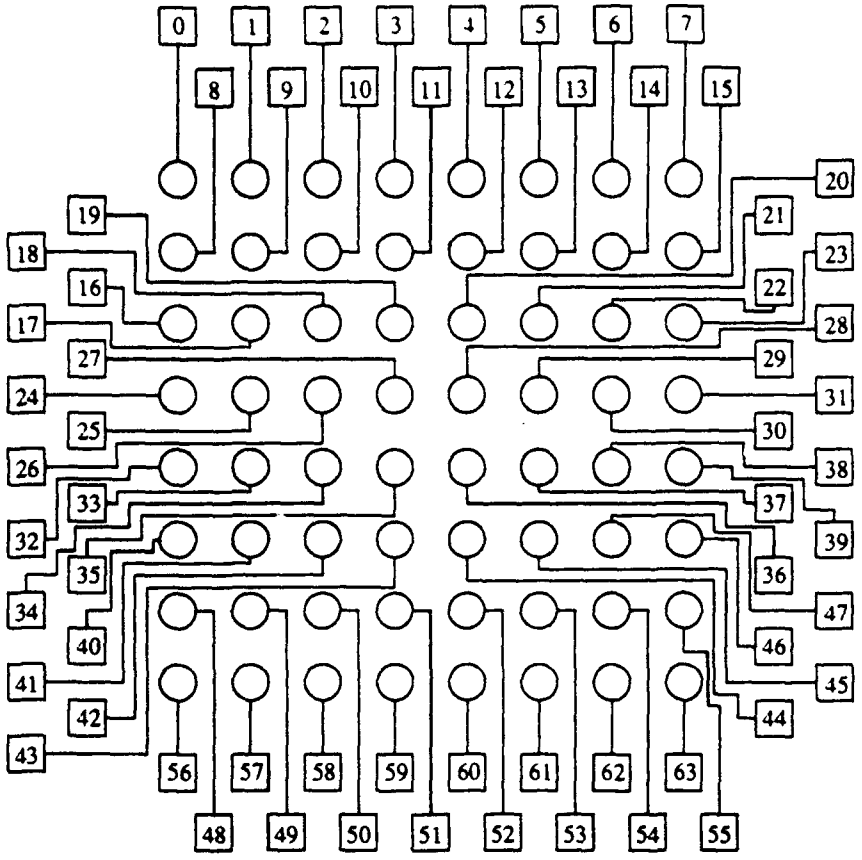


Figure 25: *Array of individually addressable VCSELs. Lead geometry shown is schematic – not actual.*

atoms in the MQW region are energized electrically. A small amount of light is generated through spontaneous emission, and a light wave traveling through the amplifying medium interacts with an energized atom. Stimulated emission occurs, and the atom converts its energy to light at the same wavelength as the traveling wave. Partially reflective mirrors (dielectric stacks in a typical implementation) at the ends of the VCSEL allow only a fraction of the light to pass, while the remainder of the light remains within the cavity to continue the process of amplification.

5.2 VCSEL Configurations

VCSELs are manufactured in three primary configurations: (1) individually addressable, (2) matrix addressable, and (3) linearly addressable. A schematic of an 8×8 individually addressable array is shown in Figure 25. Each VCSEL has a ground (n) terminal and a positive (p) terminal. All of the VCSELs share the same ground, but a separate p contact is provided for each laser. An 8×8 array thus requires 64 p contacts, which are indicated by the numbered bonding pads at the edges of the array. For small arrays, individual addressing may work well as long as the number of bonding pads is not greater than the number of pins on a typical chip carrier, which is on the order of just a few hundred. We initially considered using 8×8 arrays in the system, which requires 64 pins on each chip carrier. This is a reasonable degree of complexity, but the complexity quickly grows as we scale the system to larger sizes, and so we considered alternative configurations as well.

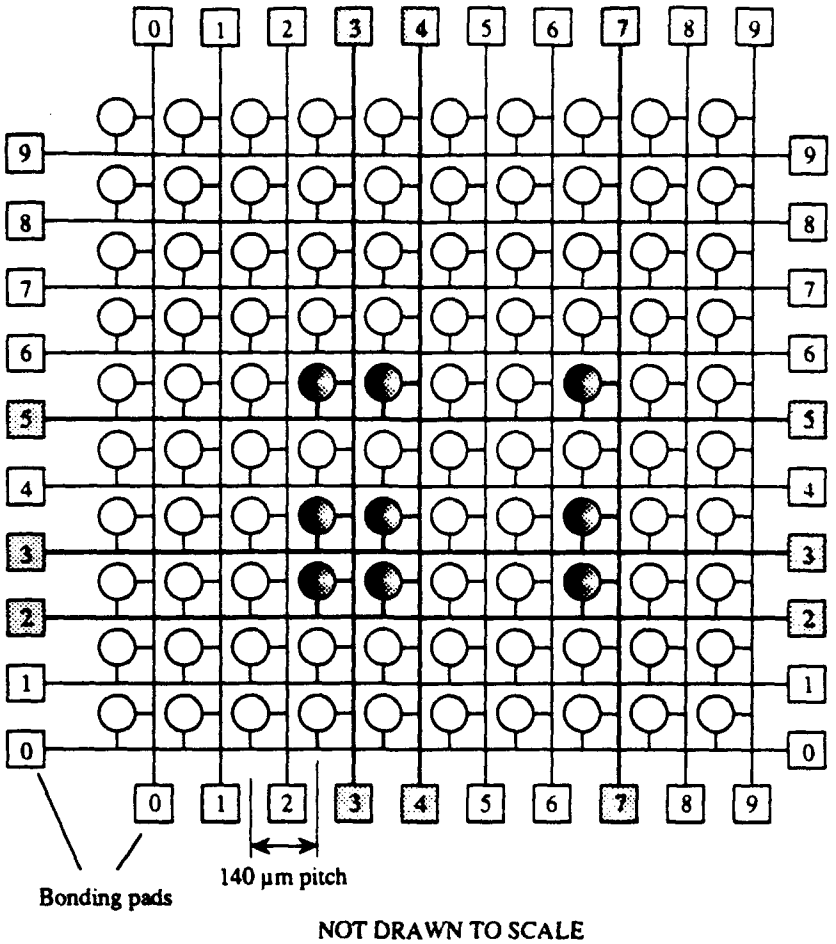


Figure 26: *A matrix addressable array of devices requires only a linear growth in the number of bonding pads for a power of two growth in the number of devices. The indicated pattern is selected by enabling rows 2, 3, and 5 and enabling columns 3, 4, and 7.*

Figure 26 shows a schematic of a 10×10 array of VCSELs loaned to us by AT&T. The VCSELs are in the center portion, and the bonding pads are located at the edges. The pitch of the lasers is 140 µm. which is the same as the pitch of the bonding pads. Each row of VCSELs shares the same ground. which has two electrically common bonding pads at the left and right of the array. For the 10 rows shown in the figure, there are 10 independent n lines, which are each connected to two distinct bonding pads. The p lines are connected to the columns in a similar manner, and so there are 10 independent p lines, which connect the p contacts of the 10 VCSELs in a column. A VCSEL must have power applied to both its n and p contacts. To enable a VCSEL at location (i, j), in which i identifies a row and j identifies a column, the corresponding i row and j column bonding pads must be powered. The n voltage is applied to the row pad and the p voltage is applied to the column pad. If a voltage is applied to more than one pad, then the corresponding collection of VCSELs is enabled. In Figure 26, power is applied to rows 2, 3, and 5 and columns 3, 4, and 7, which enables the nine VCSELs at the corresponding crosspoints.

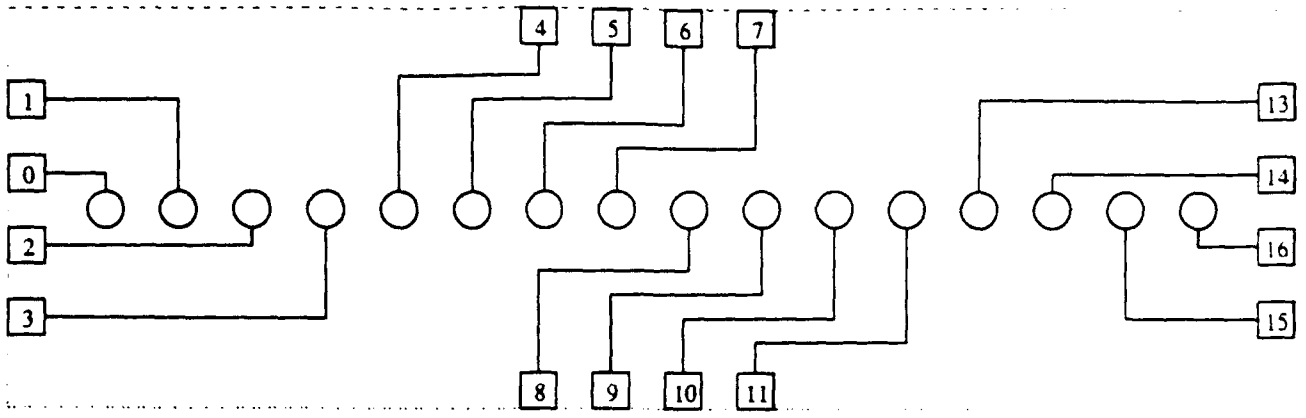


Figure 27: *Array of linearly addressable VCSELs.*

An advantage of the matrix addressable configuration is that for an N^2 increase in the size of an array, the bonding pad complexity increases by only $2N$, which allows for a simplified electronic interface. A disadvantage is that the computer designer loses a degree of freedom in selecting combinations of logic gates to enable or disable. For example, in Figure 26, there is no combination of active rows and columns that will generate a checkerboard pattern. Despite the limited number of possible on/off combinations for a matrix addressable array, the complexity of the optics and the complexity of the electronic addressing are simplified, which are currently more important considerations.

A third configuration of VCSELs is shown in Figure 27, in which the VCSELs are organized in a single line, and are individually (linearly) addressed. In this configuration, a line of VCSELs can be very long, which presents a packaging problem because chip packages are rectangular. For this reason, the bonding pads are organized in the outline of a rectangle as shown. The linear configuration, in conjunction with a spot-array generation technique, may provide an alternative method of array illumination. For the effort reported here, we did not plan on using this approach since we had large enough 2D arrays for our needs. As the system complexity grows, however, we may need to consider a combined linear array/spot-array generation technique, as discussed in the next section.

5.3. Microlasers vs. Spot-Array Generation

The individually addressable VCSEL configuration is the most flexible arrangement, and so we planned the system around this configuration. We used individually addressable VCSEL arrays provided by Bandgap are used for presetting the S-SEED device array, but these particular devices did not support the wavelength needed to perform readouts. We anticipate that VCSELs matching the S-SEED wavelength will be available in the near future, and for that reason, we investigated the use of a single VCSEL array for performing S-SEED presets, readouts, and masking.

We found that there are a number of benefits in using microlasers as opposed to a non-addressable spot-array generation technique (such as Dammann gratings or cascaded calcite slabs). For example:

1) The masks may sometimes be mapped into the individual microlaser modulation. This is a degree of freedom that is now provided with the addressable microlaser arrays.

2) If one microlaser is used to drive each device window, presets can be accomplished in a spatially varying pattern by simply turning on one or the other of a pair of microlasers for a given S-SEED.

3) The output of each microlaser can be "trimmed" to balance desired uniformity requirements through the system.

4) Coherence effects are reduced since each microlaser can be incoherent with respect to its neighbor. Thus it is not necessary to maintain orthogonal polarizations on the overlapping input spots on a given SEED window.

5) The power scalability is improved since the sources can be distributed across an extended plane which continues to supply its own light. When a single source is distributed with an array generator such as a Dammann grating, more power is required in the single source as the arrays are scaled.

A possible disadvantage of using microlasers to read out the states of the devices (as power beams) is that any temporal skew in the turn-on times of the lasers can result in erroneous spurious switching in the S-SEEDs. We were not able to test this experimentally during the reported period of work, however, we believe that this skew can be easily controlled at low switching speeds since the "time sequential gain" property of the S-SEEDs filters out spurious skews. This is an important property for simplifying the testing of the optics, but the skew issue may still pose a problem at high data rates. For this reason, we considered a future optical approach that does not use latching devices. Alternative "CELL" devices might then be used, as discussed in the next section.

5.4. Microlaser Development and Applications

Early in the reported contract period, microlaser pioneer Dr. Jack Jewell left AT&T Bell Laboratories to join Photonics Research Incorporated, which specializes in microlaser devices. During Jewell's transition period he joined our ES&E effort.

At Rome Laboratory, Jewell interacted initially with James Battiato and Thomas Stone on the optical processor effort. After learning of Stone's use of calcite plates to generate arrays of light beams, Jewell proposed using quarter-wave plates between the calcite plates in order to pattern the beams in a more rectangular fashion. Jewell had used this approach at Bell Laboratories with expensive quartz quarter-wave plates and Wollaston prisms to generate square and rectangular arrays of beams. For the Photonics Center effort, Jewell brought inexpensive thin-film polymer waveplates to the laboratory. Besides their low cost, the thin-film polymer quarter-wave plates are compact and rugged, and can be cemented between the calcite plates to form a compact monolithic array generator. There was not enough time, however, between Jewell's entry to the project and the departures of personnel Stone and Battiato to test this configuration.

Jewell assisted Stone and Battiato in the final demonstration of a calcite beam array generator during the latter part of the Summer of 1991 when all three were at Rome Laboratory. An array of (16) beams was generated with (4) calcite plates. The demonstration required accurate rotational alignment of the plates in three orientational axes. Longitudinally, the calcite plates had to be placed as close together as possible in order to minimize the effect of the beams walking off at different angles and

therefore being clipped by limiting apertures further down in the optical system. A monolithic cemented-together system, either with or without the quarter-wave plates, may overcome most of these problems.

In a proposed configuration, the S-SEED processor will need data to be fed into it in the form of a one-dimensional array of on/off light beams, which in turn receive data electronically. Jewell investigated this configuration using a one-dimensional array of VCSELs with a 25 μm pitch. This particular VCSEL sample was fabricated at Sandia National Laboratory and was brought to Rome Laboratory by Jewell. This sample was of an early design and required approximately 5 V and 5 mA to reach lasing threshold. Furthermore, the VCSELs could not be wire-bonded, and so they were activated by an electrical probe tip. Despite these shortcomings, the VCSELs were operated in a continuous-wave (CW) mode at room temperature.

Since it was not practical to insert many probe tips into the experiment, Jewell connected six adjacent VCSELs in the array by using electrically conducting silver epoxy. Despite the high power requirements and close spacings, the VCSELs operated simultaneously up to about a 30-40% duty cycle. Similar devices on an 80 μm pitch, as envisioned for the S-SEED processor, would be able to operate CW simultaneously. VCSELs of a modestly improved design would operate simultaneously even in such a closely spaced (25 μm or less) array continuously at room temperature, allowing for miniaturization of the processor. Partially fabricated VCSELs of a much more advanced low resistance design were also tested in the Photonics Center using a Hewlett Packard parameter analyzer. The current *vs.* voltage characteristics of these devices showed greatly decreased resistance and withstood 40 mA continuous current in 10 μm diameter devices.

The suitability of VCSEL structures to function as optical detectors under reverse bias was also investigated in Dr. Michael Parker's laboratory (Rome Laboratory) with the same VCSELs that were tested for suitability in an input array. In order to perform the tests, Jewell had to break devices from the corner of the VCSEL chip. This "detector chip" was then separately mounted and reverse biased. Since a reverse biased VCSEL forms a resonant detector, it is essential that the light entering it is of the appropriate wavelength corresponding to the cavity resonance. A tunable dye laser or titanium sapphire laser is ideal for this application. Unfortunately, we did not have a titanium sapphire laser available to us at the time so we could not test the VCSELs as detectors. The most appropriate source available was the array of VCSELs on the original chip. Although the VCSELs are not tunable in wavelength, they lase at the cavity resonant wavelength which closely (but not exactly) matches the resonant wavelength of the detectors.

There are several reasons why the wavelengths of the lasers and detectors do not match perfectly. First, due to thickness nonuniformity in the epitaxial growth of the VCSEL wafer structure, the resonant wavelength varies with position on the wafer. Second, the laser emission departs slightly from the original cavity resonance due to optimum gain/loss considerations and changes in the cavity refractive index caused by heating and carrier injection. Third, the reversed-biased VCSEL structure has its resonance tuned via the quantum-confined stark effect (*i.e.* the same effect used in the S-SEEDs). With all of these effects occurring simultaneously, it is highly unlikely that optimum matching of the laser emission wavelength and the detector resonant wavelength will occur. If the wavelengths are too greatly mismatched, essentially no detector response is seen. Nonetheless, a detector responsivity of greater than 0.1 Amps/Watt was observed in the short time there was to

perform the measurements. The theoretical maximum responsivity is about 0.7 Amps/Watt. Considering the likelihood of wavelength mismatch, imprecise focusing, and lack of time, the results are very encouraging.

At Rutgers University, Jewell investigated VCSEL based approaches to the optical processor that are most likely to affect future work. A two-dimensional array of individually addressable VCSELs was proposed to create input data or as an arrayed power source for the S-SEEDs. Even if the two-dimensional VCSEL array is smaller than the processor array, using an *array* of sources rather than only one can greatly reduce the beamsplitting required to produce the final beam array. For example, to power a 64×64 array, 2^{12}=4096 beams are required. Thus 12 binary splits are needed in the calcite plate spot-array generator. The use of a 16×16 VCSEL array would reduce the number of binary splits to only four, and the splits would only be at small angles.

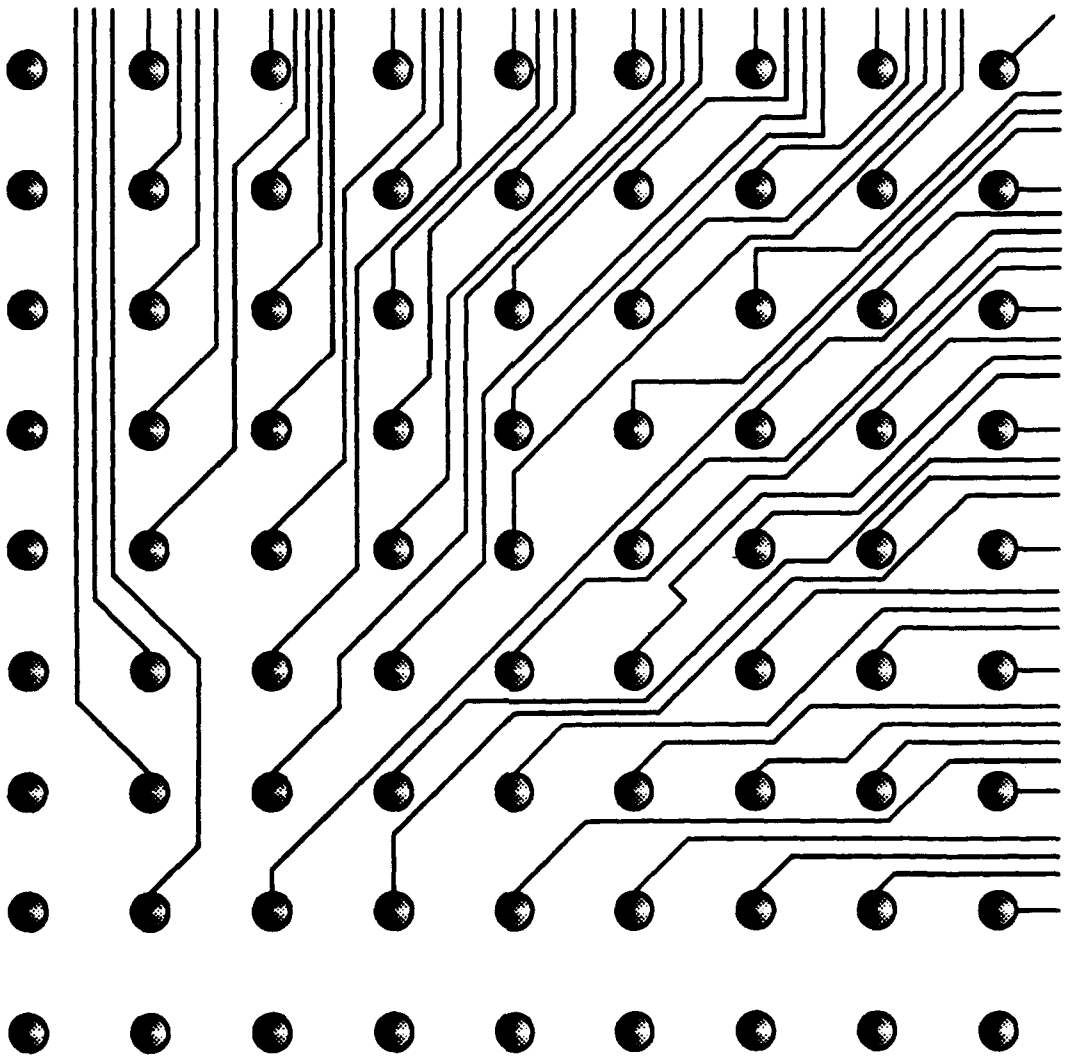


Figure 28: *Lead pattern for one quarter of a 16 × 16 individually addressable VCSEL array.*

Getting contact leads to all of the lasers in such an array in a uniform manner is a complex problem, and Jewell first mapped out a pattern for an 8×8 array (64 contact leads) where in some areas two leads must pass between adjacent VCSELs. Leads that address the centermost VCSELs are longest, and so these leads are wider in order to equalize the resistances in all of the leads. Jewell also laid out a preliminary wiring pattern for a 16×16 array having 256 leads with as many as four leads passing through adjacent VCSEL elements. The lead pattern illustrated in Figure 28 is the result of several iterations in which the wiring density is made as uniform as possible. Figure 28 shows only one quarter of the array, for clarity. The pattern can be duplicated and rotated 90° three additional times in order to generate the entire lead pattern.

Jewell also proposed a scheme for accomplishing reconfigurable interconnects based on addressing arrays of surface emitting laser logic (CELL) [5, 6] devices. The concept makes use of the optical-in/optical-out basic logic operation of the CELLs, then additionally implements electrical addressing to control the functionality of the CELL. For example, an electrical bias at one level might result in AND operation of the CELL, while a higher voltage bias could produce OR behavior. An array such as that shown in Figure 28 could thus have a completely arbitrary and completely reconfigurable arrangement of AND and OR gates. For typical optical processing architectures, however, a completely arbitrary capability is not necessary. For example, alternating rows of AND and OR gates may be appropriate, which is much simpler than supporting individual selectivity of the logic functions. For these cases a single contact can address each row of the array, which simplifies the problem of scaling up the system to a large size.

Some of Jewell's work was directed toward improving the efficiency of the VCSELs, specifically in developing lower-resistance devices. The fabrication portion of this work was performed in collaboration with Bellcore, and produced the partially fabricated low-resistance devices which were tested at Rome Laboratory. This work may increase the speed of addressability for input arrays and improve the operating speed of CELLs. Reduced heat generation will also allow larger numbers of devices to be fabricated in smaller areas.

6. AN APPLICATION IN RECONFIGURABLE MEMORY

Consider again the circuit shown in Figure 5. In the Photonics Center S-SEED processor, the outputs are disabled by selectively disabling microlasers. The goal of reconfiguration here is to modify the decoder so that different words respond to the same address at different times. The interconnect pattern shown in the circuit assigns address a_0a_1=00 to d_0, a_0a_1=01 to d_1, a_0a_1=10 to d_2, and a_0a_1=11 to d_3.

A method for freely moving arbitrarily sized objects through memory is important for graphics applications involving images, matrix operations such as the transpose, and database mining. Computer performance increasingly degrades as the size of a moving data object increases. In various graphics applications, a large background remains motionless, and objects that move are kept very small and are maintained in separate buffers that are mixed into the video stream. A few techniques can be applied to give the illusion of large scale motion, such as scrolling the color map to give the impression of wave motion, but large sections of memory do not move quickly. By exploiting the gate-level reconfiguration aspect of our S-SEED processor, however, arbitrarily sized objects may be moved as quickly as fixed size small objects can be moved. The concept is to reconfigure the

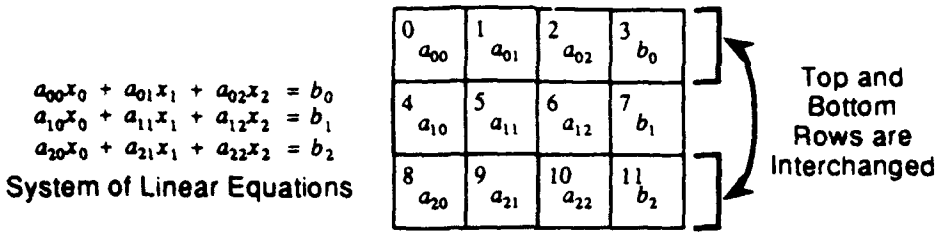


Figure 29: *An augmented matrix for a system of linear equations in three unknowns.*

crosspoints of the address decoder of the RAM in parallel, in order to achieve the logical effect that data has moved.

A potential opportunity for this type of reconfigurable optical memory is the application of Gaussian elimination to the solution of linear equations, which is important for controlling a phased array radar system. The process is data-independent if the problem of pivoting, which involves rearranging rows so that the top left element of each submatrix is relatively large, is ignored. However, in the real world, zeros or very small numbers do in fact appear along the diagonal, so that the pivoting problem must be addressed, possibly through interchanging rows.

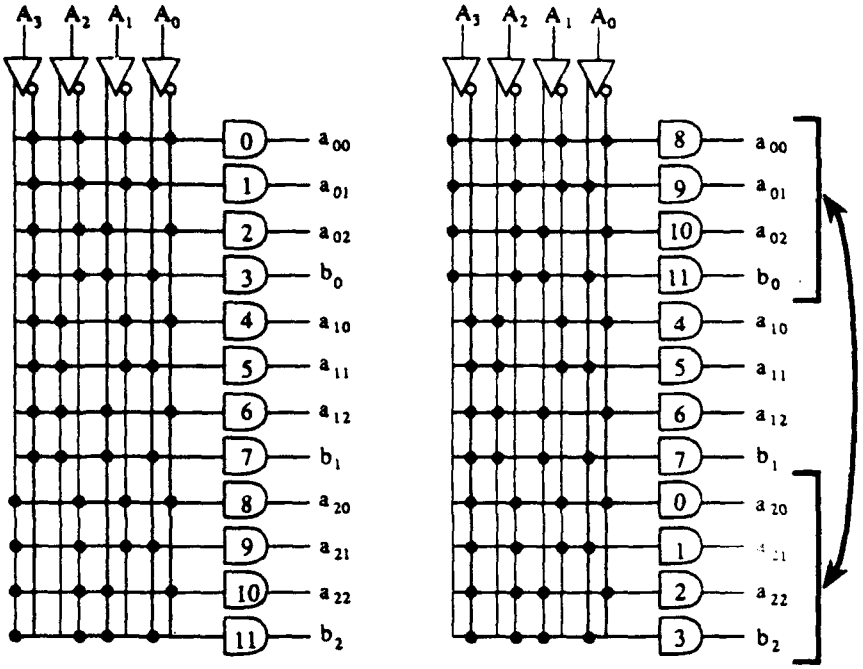
A reconfigurable interconnection technology can offer a solution without compromising performance severely by simply reconfiguring the decoder section of the memory that stores the rows of the matrix. For example, consider the augmented coefficient matrix shown in Figure 29 for three linear equations in three unknowns. The indices in the upper left corners of the twelve cells indicate the addresses of the memory locations that store the corresponding coefficients.

Figure 30 shows two decoder circuits for a memory that maps four-bit addresses into spatial locations. The circuit on the left is a conventional decoder that maps addresses into locations according to the matrix layout shown in Figure 29. The circuit on the right shows the configuration of a decoder that swaps the top and bottom rows of the matrix by changing the crosspoint settings through a reconfigurable interconnection approach. Notice that the actual data has not moved. Only 1/6 of the crosspoints are changed between the two forms, even though 3/4 of the elements are interchanged. Further, for a modest word size of 32 bits, the total number of bits that are effectively interchanged are $3/4 \times 12 \times 32$ bits = 288 bits even though only 16 crosspoints are changed in the decoder. An important property of this approach is that the modified decoder is simply projected into the system without regard for the actual data being interchanged, so that explicit data paths between all possible pairs of rows that might be interchanged do not need to be provided. This effect is more pronounced for complex interchange operations such as a transpose, in which every element is affected. A single pattern that is imaged into the decoder in a single step is all that is required to implement the transpose.

7. SOFTWARE

7.1 The Ramgen RAM Generator

The RAM designs shown in Figures 5 and 6 were created manually rather than using design tools because they are tightly constrained and very small. However, on a scaled up system, we anticipate



Generic address decoder

Customized address decoder interchanges first and third rows

Figure 30: *Two forms for a four-variable address decoder.*

relying heavily on tools developed both at Rutgers University under joint AFOSR/ONR support and at TIS Inc. under Rome Laboratory support.

A generic optical RAM design developed by Murdocca and Sugla [7] was coded in software at TIS Inc. under Air Force contract F30602-91-C-0101. The program is called Ramgen, which translates an input specification in terms of word size and the number of words into a PostScript graphic output showing all of the connections for a banyan style of interconnect. The output is broken onto several pages for large designs. A sample of the output of the program is shown in Figure 31 for a 16-word by two-bit (dual-rail) RAM.

7.2 The Xopid Interactive Circuit Design Tool

During the exploratory phase of the RAM design, we made use of an X-windows Optical Programmable Interactive Design tool (Xopid) that uses the X graphical interface. The Xopid tool allows logic gates to have fan-ins and fan-outs that vary, and allows circuits to have irregular interconnections between gates and between higher level structures such as PLAs. These features allow us to study the trade-offs involved when fan-in/fan-out values higher than two are used and when connections are not constrained to being topologically equivalent to the perfect shuffle, banyan, or crossover.

In order to manage the complexity of circuit design for regular interconnects, the interactive design tool makes use of a collision detection strategy that guides the design process so that signals do not collide as a result of using common paths, which is a situation commonly referred to as **blocking** in

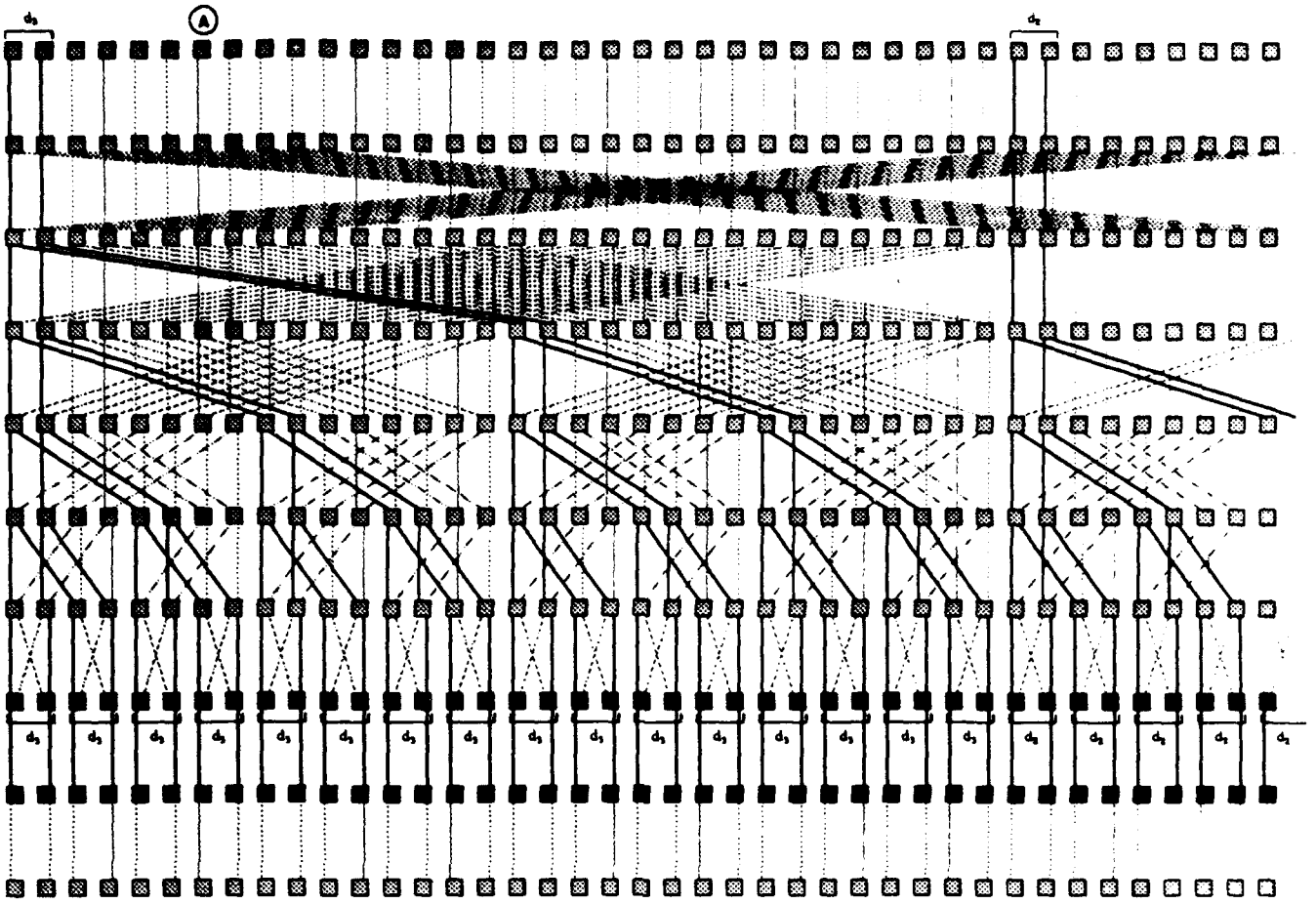


Figure 31: *Sample output from RAM generator.*

switching applications. Collision detection is a significant issue here because the physical circuit layouts and their functional behaviors are tightly coupled.

In more detail, Xopid is a menu-driven tool that allows the user to draw and manipulate digital circuits interactively in an X window. The user interface to Xopid is shown in Figure 32. Five vertically stacked windows comprise the display area: the **command window**, the **file-label window**, the **main drawing window**, the **help window**, and the **message window**. The command window contains control buttons that the user selects for different circuit manipulation operations. When a button is selected, it is highlighted and a brief message describing its function is displayed in the help window. The main drawing window displays the circuit that is being designed. The virtual drawing area is larger than the main drawing window, which displays a portion of the virtual drawing area. The main drawing window can be moved over the virtual drawing area by using the scrollbars, which also indicate the relative sizes of the main drawing window and the virtual drawing area. If the execution of a user command results in an error or some other exceptional behavior, a message is displayed in the message window. The file-label window displays the name of the circuit being manipulated.

A synopsis of the functions available to the user is given below. Any command that ends with an ellipsis (...) indicates that a dialog box appears that prompts the user for more information.

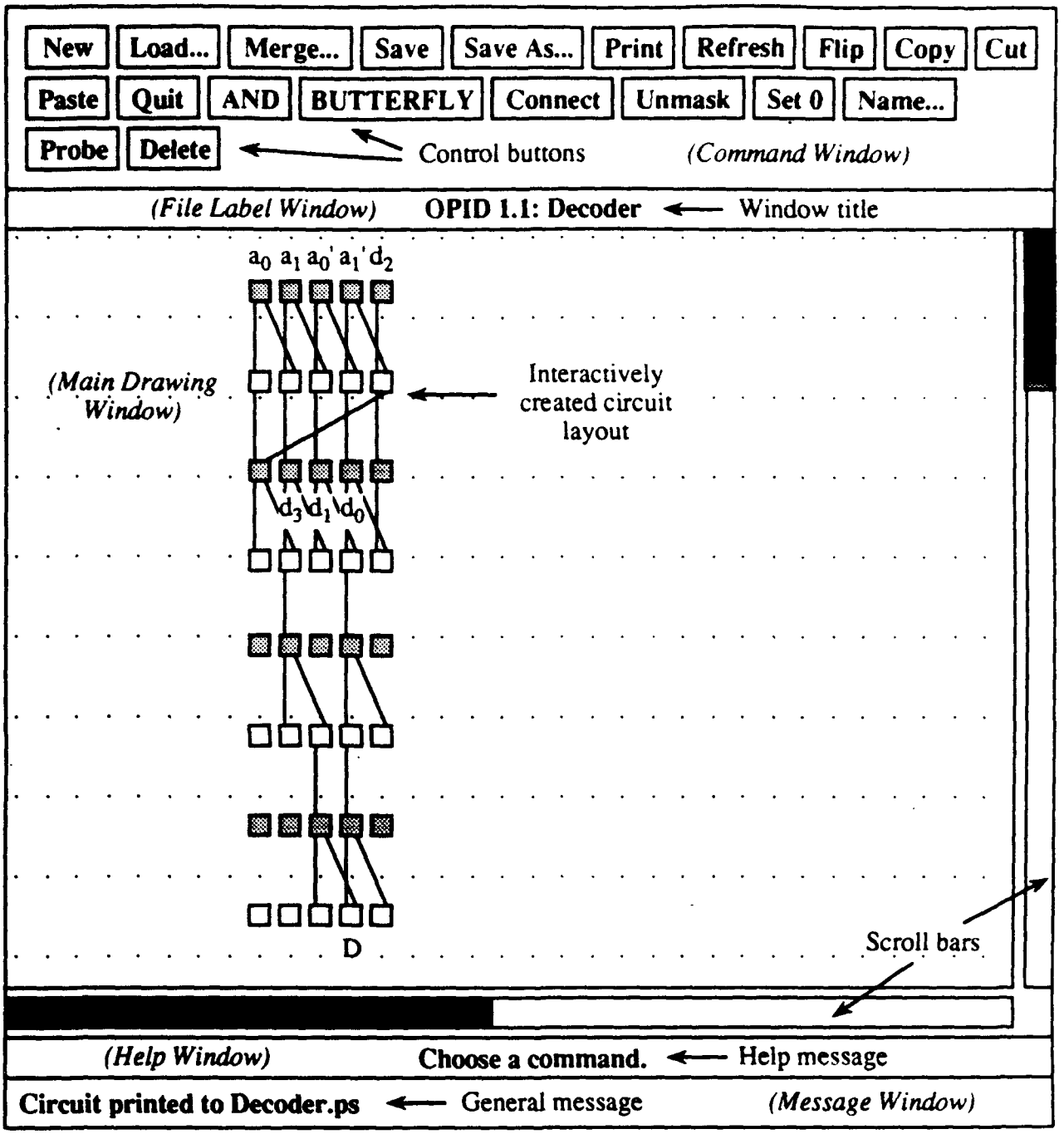


Figure 32: *The Xopid user interface. The decoder circuit from Figure 5 is shown in the Main Drawing Window.*

NEW Clears the current circuit.

LOAD... Prompts the user to specify a `.cir` file (a circuit file stored in Xopid format, with the filename extension '`.cir`'). The circuit described in this file then becomes the current circuit. If the specified file does not exist, an empty circuit becomes the current circuit.

MERGE... Prompts the user to specify a `.cir` file. The circuit in this file is merged into the current circuit at a position that the user selects with the mouse. The merge operation fails if a circuit overlap situation exists.

SAVE Saves the current circuit in the file named by extending the filename displayed in the file-label window with a `.cir` extension.

SAVE AS... Prompts the user to specify a `.cir` file. The current circuit is then saved in this file.

PRINT Prints the current circuit in PostScript format to the file named by extending the filename displayed in the file-label window with a `.ps` extension.

REFRESH Redraws the circuit on the bitmap that is displayed in the main drawing window.

FLIP Waits for the user to specify a rectangular region by depressing the left mouse button on the upper left corner of the region, dragging the pointer to the lower right corner of the region and then releasing the button. A copy is made of the sub-circuit corresponding to the user-specified rectangular region, which is flipped along a vertical axis passing through the center of the region and stored in a file named `.Clipboard.cir`, from where it can be pasted using the PASTE option. This operation is useful in building crossover circuits from smaller crossover circuits, which are symmetric about a vertical axis.

COPY Similar to FLIP except that the sub-circuit is not flipped before it is stored in `.Clipboard.cir`.

CUT Similar to COPY but deletes the sub-circuit corresponding to the user-specified region from the current circuit.

PASTE Waits for the user to specify a position with the mouse, which is where the upper left corner of the circuit stored in `.Clipboard.cir` is merged into the current circuit, providing the operation does not result in an overlap.

QUIT Exit from Xopid, discarding the current circuit.

OR/NOR/AND/NAND Waits for the user to specify a rectangular region (as described in FLIP) and fills the rectangular region with logic gates of the type displayed in the help window. If a gate already exists in the region, its type is changed to that displayed in the help window. The user can toggle through the gate types by repeatedly selecting this command button.

BUTTERFLY/SHUFFLE/CROSSOVER Waits for the user to specify a rectangular region (as described in FLIP) and inserts connections corresponding to the current interconnection pattern between gates in this region. The user can toggle through the interconnection patterns by repeatedly selecting this command button. Note: the terms "butterfly" and "banyan" are used interchangeably here.

CONNECT/DISCONNECT Waits for the user to depress the left mouse button over a gate, drag the pointer till it is over another gate, and then release the button. If the CONNECT option is active, a new connection is made between an output of the first gate and an input of the second gate if one does not already exist. If the DISCONNECT option is active, the existing connection, if any, between an output of the first gate and an input of the second is removed. The operation

is performed between successive rows only. The active option is displayed in the help-window. The user toggles between the two options by selecting this command button.

MASK/UNMASK Waits for the user to specify two gates (as described in CONNECT/DISCONNECT). If the UNMASK option is active, a path of connections, if one exists, leading from the output of the first gate to the input of the second gate is found and all connections on this path are unmasked (that is, connections are enabled). If the MASK option is active, all connections on the path are masked (disabled). The active option is displayed in the help window. The user toggles between the two options by selecting this command button.

SET 0/SET 1/UNSET Waits for the user to select a gate. If the SET 0 option is active, the output of the selected gate is set to 0. If the SET 1 option is active, the output of the selected gate is set to 1. If the UNSET option is active, any Boolean value to which the output of the selected gate had been tied is removed. The active option is displayed in the help window. The user toggles between the options by selecting this command button.

NAME... Prompts the user to specify a name for a gate and waits for the user to select a gate. The output signal of the selected gate is then given the specified name. If a name is not specified, and if the output of the gate already has a name, that name is removed.

PROBE Waits for the user to select a gate. The output value generated at the gate and the Boolean expression representing the gate's output are displayed in the message window.

DELETE Waits for the user to specify a rectangular region as described in FLIP. All gates that lie within this region are deleted from the current circuit as well as all connections that are incident on any gate in the region.

7.3 MSI Shuffle Placement Software

A software tool was used for mapping MSI component inputs and outputs onto a single perfect shuffle stage. The placement program reads an input file that describes a number of MSI components that form a network. The program generates (if possible) a side-by-side placement of the components such that a single perfect shuffle satisfies all connection requirements.

A sample input file is shown below, in which there are eight components (boxes) that are to be mapped onto a 32-wide perfect shuffle. A time-limit stopping point of 1025 trials is used.

```
32          The width of the perfect shuffle
8           The number of boxes
4           Width of box #0
4           Width of box #1
4           Width of box #2
4           Width of box #3
4           Width of box #4
4           Width of box #5
4           Width of box #6
4           Width of box #7
7           Number of connections
0           1     First connection is from box #0 to box #1
1           2
2           3
```

```
3           4
4           5
5           6
6           7
1025              Number of orderings to try
```

The corresponding output file is shown below. Trial 43 shows the ordering of components that results in a successful mapping. The program could have stopped at this point, but we allowed it to continue until the time-limit in a search for alternate solutions. No alternate solutions were found within the time limit.

```
Starting time = Thu Mar 11 08:38:15 1993

The width of the shuffle is 32
The number of boxes is 8
The box widths are:
box_widths[0] = 4
box_widths[1] = 4
box_widths[2] = 4
box_widths[3] = 4
box_widths[4] = 4
box_widths[5] = 4
box_widths[6] = 4
box_widths[7] = 4
The number of connections is 7
The net-list, in the form from --> to:
0 --> 1
1 --> 2
2 --> 3
3 --> 4
4 --> 5
5 --> 6
6 --> 7
The number of orderings that will be tried is 1025
7 connections need to be made

TRIAL 0 Ordering: 0 1 2 3 4 5 6 7 -> 2 connections made
TRIAL 1 Ordering: 0 1 2 3 4 5 7 6 -> 3 connections made
TRIAL 5 Ordering: 0 1 2 3 4 7 6 5 -> 4 connections made
TRIAL 17 Ordering: 0 1 2 3 6 7 5 4 -> 5 connections made
TRIAL 29 Ordering: 0 1 2 4 3 7 6 5 -> 6 connections made
TRIAL 43 Ordering: 0 1 2 4 7 3 6 5 -> 7 connections made
TRIAL 1024 Ordering: 0 2 4 5 6 7 1 3 -> 0 connections made

            *** END OF RUN ***

Ending time = Thu Mar 11 08:38:40 1993

Elapsed time = 25 seconds
Total number of samples tried: 1025
```

Best oraering provides 7 out of 7 needed connections

8. CONCLUSION

Two RAM designs were created for an all-optical processor composed of arrays of VCSEL controlled S-SEED devices. Both designs make use of low fan-out split-and-shift interconnects, which are customized by selectively disabling VCSELs. A number of architectural issues were explored as a result of this configuration. The problems of X-Y and Z folding involve mapping a circuit with arbitrary dimensions onto a physical system that has fixed dimensions. In the Z dimension, a perfect shuffle turns out to be the most flexible interconnect, whereas the split-and-shift is one of the most practical.

The VCSEL controlled S-SEED configuration allows the system to be dynamically reconfigured during operation. A potential opportunity for this approach may be the solution of systems of linear equations for a phased array radar application. This can only be effective if large arrays of VCSELs can be fabricated, and some exploratory VCSEL work was performed with regard to lead patterns for a 16×16 VCSEL array.

Our participation in the Photonics Center S-SEED processor project has led to a working prototype of the target system. The work reported here was performed on an ES&E effort that spans the middle portion of the history of the processor's development. A manuscript detailing the working system is currently in preparation.

9. REFERENCES

[1] Lentine, A. L., H. S. Hinton, D. A. B. Miller, J. E. Henry, J. E. Cunningham and L. M. F. Chirovsky, "The symmetric self electro-optic effect device," in *Conference on Lasers and Electro-optics*, Technical Digest Series 1987, vol. 14, (Optical Society of America, Washington, D.C., 1987, 249), postdeadline paper.

[2] Jewell, J. L, A. Scherer, S. L. McCall, Y. H. Lee, S. J. Walker, J. P. Harbison and L. T. Florez, "Low threshold electrically-pumped vertical cavity surface-emitting microlasers," *Electron. Lett.*, **25**, pp. 1123-1124, (1989).

[3] Prise, M. E., N. C. Craft, M. M. Downs, R. E. LaMarche, L. A. D'Asaro, L. M. F. Chirovsky, and M. J. Murdocca, "An Optical Digital Processor Using Arrays of Symmetric Self-Electrooptic Effect Devices," *Applied Optics*, **30**, no. 17, pp. 2287-2296, (Jun. 10, 1991).

[4] Murdocca, M. J., A. Huang, J. Jahns, and N. Streibl, "Optical Design of Programmable Logic Arrays," Applied Optics, 27, pp. 1651-1660, (May 1, 1988).

[5] Olbright, G.R., R. P. Bryan, K. Lear, T. M. Brennan, G. Poirier, Y. H. Lee, and J. L. Jewell, "Cascadable Laser Logic Devices: Discrete Integration of Phototransistors with Surface-Emitting Laser Diodes," *Electronics Letters*, **27**, 216, (1991).

[6] Olbright, G. R., R. P. Bryan, T. M. Brennan, K. Lear, G.E. Poirier, W. S. Fu, J. L. Jewell, and Y. H. Lee, "Surface-Emitting Laser Logic," *OSA Proceedings on Photonic Switching*, H. Scott Hinton and J. W. Goodman, eds., **8**, 247, (1991).

[7] Murdocca, M. J. and B. Sugla, "Design for an Optical Random Access Memory," *Applied Optics*, **28**, pp. 182-188, (Jan. 1, 1989).

MISSION

OF

ROME LABORATORY

Rome Laboratory plans and executes an interdisciplinary program in research, development, test, and technology transition in support of Air Force Command, Control, Communications and Intelligence (C3I) activities for all Air Force platforms. It also executes selected acquisition programs in several areas of expertise. Technical and engineering support within areas of competence is provided to ESC Program Offices (POs) and other ESC elements to perform effective acquisition of C3I systems. In addition, Rome Laboratory's technology supports other AFMC Product Divisions, the Air Force user community, and other DOD and non-DOD agencies. Rome Laboratory maintains technical competence and research programs in areas including, but not limited to, communications, command and control, battle management, intelligence information processing, computational sciences and software producibility, wide area surveillance/sensors, signal processing, solid state sciences, photonics, electromagnetic technology, superconductivity, and electronic reliability/maintainability and testability.