

In Proceedings of the IEEE Special Issue on Optical Computing, Nov. 1994.

Optoelectronic Time-of-Flight Design and the Demonstration  
of an All-Optical, Stored Program, Digital Computer

Harry F. Jordan, Vincent P. Heuring, Robert Feuerstein

Optoelectronic Computing Systems Center<sup>†</sup>  
University of Colorado and Colorado State University  
Campus Box 525, University of Colorado  
Boulder, CO 80309-0525

ABSTRACT

The recent demonstration of an all-optical, stored program, digital computer by our group focused on high speed optoelectronic design. It was made possible by a new digital design method known as time-of-flight design. A rudimentary, but general purpose, proof of principle computer was built, which is all-optical in the sense that all signals connecting logic gates and all memory are optical in nature.  $LiNbO_3$  directional couplers, electro-optic switches, are used to perform logic operations. In addition to demonstrating stored program operation in an optoelectronic digital computer, the system demonstrated the feasibility of the new design method, which does not use any flip flops or other bistable devices for synchronization or memory. This potentially allows system clock rates of the same order as device bandwidth.

This paper describes how the time-of-flight design method was motivated by the special properties of optoelectronic digital design. The basic principles of the method we employed will be discussed along with some of its potential advantages. The experimental work with digital optical circuits leading up to and including the stored program computer experiment will then be discussed. Finally, the future potential of time-of-flight design in high bandwidth optoelectronic systems will be discussed.

---

<sup>†</sup> The Center for Optoelectronic Computing Systems is sponsored in part by NSF grant number ECD 9015128 as part of the Engineering Research Centers Program, and the Colorado Advanced Technology Institute (CATI), an agency of the State of Colorado.

## 1. Introduction

Several potential advantages of processing information optically have driven research in optical computing. In digital computing, to which this paper is confined, some limitations in the way electronics is currently used are becoming evident. Driving capacitive and inductive loads at high speeds requires high peak power, and the associated electromagnetic radiation both dissipates power and causes interference. Rise and fall time limitations make reducing pulse width for faster operation difficult. Stray capacitance and inductance hinder exact impedance matching at fan-outs, thus increasing timing skew. Mechanical off-chip connections and the consequent high power needed for high speed pin drivers are also problems. Corresponding potential advantages of optical signal encoding come mainly from the high temporal and spatial bandwidths possible in optical transmission. Low signal dispersion results in a bandwidth of 1-10 terabits/sec even for a single glass fiber channel. Optical systems that communicate in two or three dimensions offer additional space bandwidth that can be used to obtain dense interconnections without mechanical contact. Impedance matching problems at optical signal fan-outs are essentially absent, making signal timing skew orders of magnitude more controllable than in electronics. The advantages of optical encoding for information transmission do not extend to logic and switching, where electronics still has definite advantages.

### 1.1. Feasibility

Goodman[1] examined optical interconnects from the device level from the viewpoint of power dissipation, and later with Ozaktas[2] from a higher system level perspective. Miller's[3] analysis was at the device level, focused on impedance transformation, and allowed for optical modulators as well as sources. Both concluded that the energetics of optical interconnection is superior to that of electronic interconnection for all but the shortest distances. Numerous experiments have explored the limits of optical communication bandwidth. For a single fiber channel, Haner[4] demonstrated ~100 femtosec resolution in a time compressed waveform. Modulating and switching signals at such a bandwidth is more difficult. Islam[5] has demonstrated NOR gate operation at 0.3 Terabits/s, and somewhat more practical fiber logic gates, based on the Sagnac interferometer and using physics which is potentially just as fast, have been demonstrated as switches for demultiplexing at 100 Gb/s [6]. Assisting the switching with electronics, electrooptic  $LiNbO_3$  directional couplers have been demonstrated at 40 GHz [7]. Spatial light modulators which are both fast and have large space bandwidth products are actively being pursued. Arrays of FET-SEED devices have been extensively studied recently and a single circuit has been demonstrated as capable of operating up to 250Mb/s [8], with a 128 element array demonstrated at 50 MHz [9].

### 1.2 Advantages of Time-of-Flight Design

The work described here uses only a subset of the potential advantages of digital optical computing. The key features are high single channel bandwidth, low dispersion, and good impedance matching properties which lead to precisely predictable signal transmission time. For laboratory scale systems, the dominant uncertainty in signal propagation time in fiber is the temperature dependence of the index of refraction of glass. It causes signal delay to vary by about one part in  $10^4$  per degree C [10]. This precision

compared to electronic delay uncertainty allows reliance on circuit layout to synchronize signals without bistable elements. Information can be stored in recirculating delay lines without using latches. Of course, even a timing uncertainty of one part in  $10^4$  must be compensated for in a recirculating loop, but this can be done using clock gates, discussed below. These ideas lead to the technique of time-of-flight design, in which no flip flops or other bistable elements are used and synchronization is based on controlled arrival time of signals at switching elements.

Techniques similar to time-of-flight design have been used in high speed digital electronic circuits in connection with transparent latches [11] and wave pipelining [12] [13]. Applications of wave pipelining have been primarily to feed forward combinational logic between stages of latches. On the order of 2-5 signal waves are allowed to propagate simultaneously through the logic by adding gates to short logic paths and adjusting gate drives to balance arrival times. Precise control of optical delays allows extending these techniques to circuits with feedback and significantly increasing the number of signal waves propagating simultaneously. Optical delay precision has also been used for clock distribution [14] and for interprocessor communication buses [15]. Researchers at AT&T have also used latching electrooptic devices to build all-optical sequential circuits [16].

### **1.3 Time-of-Flight Latchless Architectures**

The work described in this paper combines all-optical circuit design with time-of-flight synchronization to build synchronous, latch-free digital systems. Electrooptic switches are used, but the electronic signals are confined to individual switches. Thus the systems are all-optical in the sense that information is in optical format except at the point it is used to control a switch. We would argue that this use of optics for communication and electronics for computation employs optics and electronics each in their most appropriate realms. The feasibility of the design method has been demonstrated by the successful operation of digital optical circuits ranging from a serial counter [17] to a complete serial stored program computer [18].

### **1.4 Outline of the Paper**

The second section of this paper will develop the formal theory of time-of-flight design and describe algorithms used to realize the demonstrated experiments, as well as extensions which will be required to extend the work to high speed optoelectronic integrated circuits. The third section will describe the optical circuit environment tools and techniques developed to design and simulate the circuits, and the several experiments. The fourth section will discuss the potential for integrated optoelectronic versions of similar digital systems.

## **2. Theory of Time-of-Flight Design**

### **2.1 Basis for the Design Paradigm**

Time-of-flight design assumes a) that gate and interconnection delays are precisely controllable, b) that data propagation time along signal paths is known, and c) that path delays are adjusted in design so that two signals which interact at a logic gate arrive

simultaneously. A time-of-flight circuit contains only combinational logic and delays, as sketched in Fig. 1. The circuit is merely illustrative and performs no useful function. Heavy arrows represent pulse mode signals propagating through the circuit and interacting at gates. Signals derived from several consecutive sets of inputs can be moving through the circuit at a time. There are no flip flops or other bistable elements. Memory is implemented by feedback loops having precisely known delays. This is in contrast to the feedback loops internal to flip flops which have negligible delay with respect to the clock period. An explicit delay is shown as a circle labeled by the number of clock periods of length  $\Delta$  it represents. This type of design can be naturally pipelined, since precise delays prevent signals derived from a second set of inputs from overtaking those derived from an earlier set. Such systems may still be synchronous in the sense that, for any specific data bit at a point in the system, there is a master clock pulse with which it is associated. Clock gating can be used to prevent accumulation of small timing shifts in feedback loops, as discussed below.

An ideal circuit assumes memory delays are multiples of a clock period and that combinational logic is delay free, as shown in Fig. 2, drawn to correspond to the traditional sequential circuit model. Such a circuit would free run at a clock rate equal to the reciprocal of the unit delay, provided inputs were applied at the same rate. We assume

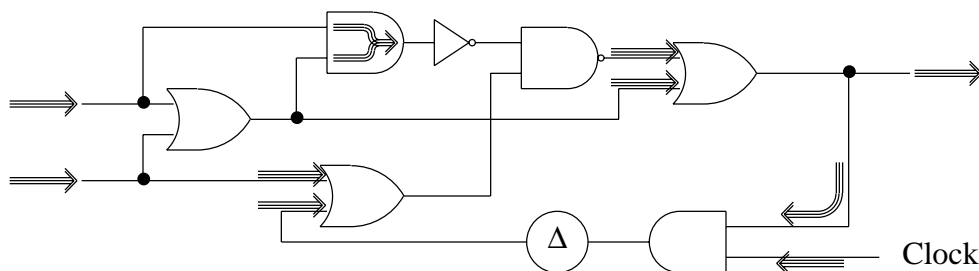


Figure 1: Concept of time-of-flight logic design

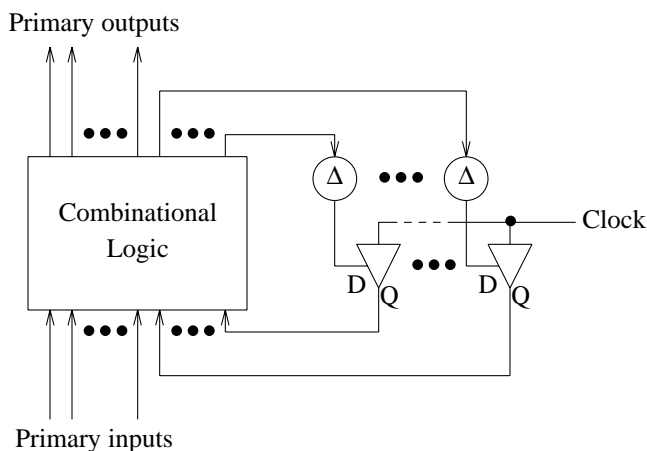


Figure 2: Time-of-flight sequential circuit abstraction

pulse mode operation. Two problems must be solved to use this design method: managing components with non-zero delay, and synchronizing such free running circuits both internally, and with external input streams. The first problem is solved for a sufficiently long clock period by reducing memory delays by the logic delay, so that the recirculation time through logic and memory is a clock period. Lumped delays are thus distributed to maintain specified delays around closed loops. Delays can also be adjusted so inputs are applied simultaneously and outputs are simultaneously available.

Synchronization is accomplished by clock gating and pulse stretching, as shown in Fig. 2, where memory elements consist of a delay and a clock gate to provide synchronization with a master clock. Although the logic performed is AND, the figure shows the clock gates as gated amplifiers to emphasize the fact that the gating operation is asymmetric. The clock gates accommodate a small arrival time uncertainty by stretching the gating pulse at D by a fraction of the duty cycle and timing it to arrive earlier than the clock leading edge by half the stretch, as shown in Fig. 3. Thus the full clock pulse is gated to Q under the worst case arrival variation in D. Not only must loops be adjusted to multiples of the clock period, but logic delays must be balanced so interacting signals arrive simultaneously at a gate.

Time-of-flight circuits have the additional feature that latency can be easily decoupled from bandwidth through pipelined, or time multiplexed, operation. For example, if pulse mode operation with less than a 50% duty cycle is assumed and the clock rate is doubled with no other change to the system, then signal pulses synchronized to odd-numbered clock pulses interact with each other exactly as did the pulses of the original system, as do signals synchronized to even-numbered clocks. The result is two time multiplexed state machines running simultaneously at the original clock rate. We have demonstrated this time-multiplexed operation in a bit serial, digital optical circuit [19].

## 2.2 Graph-Theoretic Formulation

Our network theory framework for time-of-flight design starts with standard sequential design modified by the use of delay elements for all memory. Designer added delays and reduction of specified delays are calculated from irreducible delays in real components and the circuit logic. Two influences govern realizability, one from latency (propagation time) constraints and the other from bandwidth, where component speed and delay uncertainty constrain clock rate. We first address latency constraints.

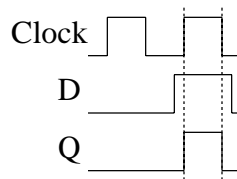


Figure 3: Clock gating with gate stretching for delay uncertainty tolerance

### 2.2.1 Latency Considerations

Consider two versions of the circuit, both having the same network graph. The first is a lumped delay design produced by a sequential logic designer, where all delays are explicitly introduced to give correct behavior and are multiples of the clock period. The second version is a circuit with the delays distributed among all components. The second circuit is deemed to operate correctly when it has the same sequential behavior as the first, perhaps with some added delay from inputs to outputs. Let  $l$  be a loop in the circuit graph. Let  $c$  be a circuit component: logic gate, fan-out, fan-in, explicit delay, or connection. Let  $b(c)$  be the idealized delay in clock periods for a component. If  $c$  is an explicit delay element  $b(c)$  is a positive integer; if it is any other component, then  $b(c) = 0$ . Since delays through different inputs and outputs of a component may vary, let  $h(c, l)$  be the unavoidable real delay through component  $c$  when it is traversed as a part of loop  $l$ . Delays are taken as negative when the loop traversal direction is opposite to signal flow.

The fastest clock feasible for a real circuit  $S$ , constrained by real non-zero delays, is found by referencing it to the ideal lumped delay circuit. A circuit  $S$  with digital operation the same as that of the lumped delay circuit is such that the real delays  $d(c, l)$  through components  $c \in S$  while traversing any loop  $l$  of  $S$  satisfy

$$\sum_{c \in l} d(c, l) = \left[ \sum_{c \in l} b(c, l) \right] \Delta t, \quad (1)$$

where  $\Delta t$  is the clock period. The delay through a real device  $c$  on loop  $l$  is the unavoidable minimum delay  $h(c, l)$  plus an additional delay introduced by the designer to satisfy (1). Practically, we can assume that additional delays are added only at interconnections. To satisfy (1), the clock period must be large enough so that for all  $l \in S$ ,

$$\sum_{c \in l} h(c, l) \leq \left[ \sum_{c \in l} b(c, l) \right] \Delta t. \quad (2)$$

Since delays are additive, it is sufficient to consider loops in a loop basis  $L$  for the graph. Figure 4 shows a circuit graph with a three loop basis and a representation of the delays on different paths through a multi-terminal network component. The integers

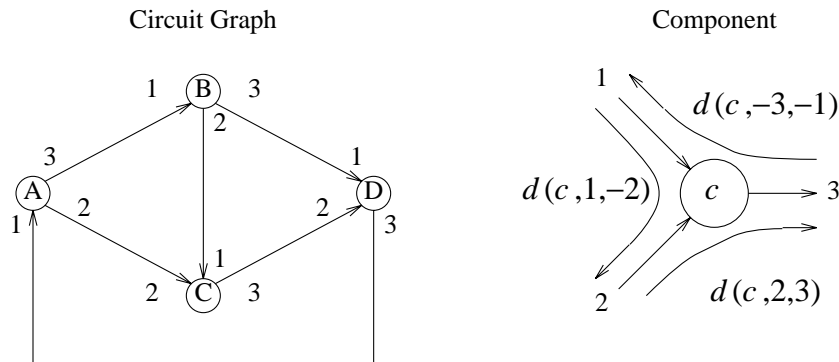


Figure 4a: A network and delays through components. Figure 4b: Signs of delays

label the input and output terminals of the components. Let the three loops of a basis for the graph of Fig. 4 be  $l_1 = ABCA$ ,  $l_2 = BDCB$ ,  $l_3 = ACDA$ . Then the loop  $l' = ABDA$  can be written as  $l' = l_1 + l_2 + l_3$ , with the delay around  $l'$  equivalent to the sum of the delays around  $l_1$ ,  $l_2$ , and  $l_3$ . A delay  $d(c, l)$  is represented as  $d(c, m_1, m_2)$  where  $m_1$  is the terminal through which loop  $l$  enters  $c$  and  $m_2$  is the exit terminal, as shown explicitly in Fig. 4b. Thus, in this example,

$$\begin{aligned} d(A, 1, 3) + d(B, 1, 3) + d(D, 1, 3) &= d(A, -2, 3) + d(B, 1, 2) + d(C, 1, -2) \\ &+ d(B, -2, 3) + d(D, 1, -2) + d(C, -3, -1) + d(A, 1, 2) + d(C, 2, 3) + d(D, 2, 3). \end{aligned}$$

Delays through a component  $d(c, l)$  can be associated with inputs and outputs by imagining an ideal point node where the logic takes place. Then delays can be written  $d(c, I_i, O_j) = dI_i + dO_j$  or  $d(c, O_j, O_k) = -dO_j + dO_k$ . Delays make a negative contribution to a path which enters an output or exits an input. Thus all circuit delays can be uniquely associated with an edge of its graph.

The problem of implementing a circuit with real delays that has the same digital behavior as the ideal circuit is then summarized by the following theorem.

**Theorem:**

Let  $S$  be a time-of-flight circuit and let  $c \in S$  be the components of  $S$ , including logic gates, fan-in or fan-out, explicit delays, and connections. Let  $L$  be a loop basis for the circuit graph of  $S$ . Let  $b(e_j)$  be the ideal delay in clock period units of the ideal lumped delays on edge  $e_j$ . Take the ideal delays to be zero for all but explicit two terminal delay devices.

A circuit with real delays  $d(e_j)$  and clock period  $\Delta t$  has an identical digital behavior to that of the ideal, lumped delay circuit provided that for all  $l \in L$

$$\Delta t \left[ \sum_{e_j \in l} b(e_j) \right] = \sum_{e_j \in l} d(e_j). \quad (3)$$

Since delays are non-negative when the graph is traversed in the direction of signal flow,

$$\Delta t \left[ \sum_{e_j \in l} b(e_j) \right] \geq \sum_{e_j \in l} h(e_j) \quad (4)$$

must hold for all loops  $l$  which traverse the edges that comprise them in the direction of signal flow.

This shows that the problem of finding the minimum the clock period  $\Delta t$  for the circuit is a constrained minimization problem with constraints given by equations (3). Given a feasible value for  $\Delta t$ , adjusting the circuit delays to give a correctly operating circuit with no excess delay is also a constrained minimization.

Although signals within an extended portion of the circuit do not settle to a stable state within the clock period, it is still possible to define a synchronous time-of-flight sequential circuit. Let master clock pulse leading edge times be  $t_i$ ,  $i = 0, 1, 2, \dots$ , so that the clock period is  $\Delta t = t_{i+1} - t_i$ . Let the circuit be represented by directed graph  $S$  with non-negative delay weights associated with edges and the master clock taken as vertex 1. A synchronous, time-of-flight sequential circuit is one in which, for any vertex  $v_j \in S$ ,

there is a reference time  $r_j$  such that the value of a logic signal at  $v_j$  at time  $t$  “refers to” or is “caused by” the master clock that occurred at time  $t - r_j$ . Informally, the  $i$ -th master clock pulse “occurs”, or “arrives at” vertex  $v_j$  at time  $\tau_{ij} = t_i + r_j$ .

All signals arise from logic operations on gated versions of the clock, so one might think of the  $i$ -th clock pulse as the one from which the inputs to vertex  $v_j$  at  $\tau_{ij}$  are derived, but logic may be done on signals referred to several different clock pulses.  $S$  must contain at least one directed path from  $v_1$  to any vertex  $v_j \in S$ , but there may be several whose delays differ by integer multiples of  $\Delta t$ . We define the reference time for vertex  $v_j$  to be the minimum total delay along any path from  $v_1$  to  $v_j$ . Signal reference times can be obtained for an arbitrary real delay circuit by solving the shortest path problem with respect to vertex  $v_1$ , which can be done in order  $n^3$  time where  $n$  is the number of vertices. The reference times will satisfy

$$r_j = \min_{k \in G_j} \left[ r_k + d(e_{kj}) \right],$$

where  $G_j$  is the set of all vertices providing inputs to  $v_j$ .

The design problem for a synchronous time-of-flight sequential circuit can now be summarized. Taking real circuit edge delays  $d(e_j)$  as the sum of an irreducible delay  $D_j$  and a designer controllable  $\delta_j$ , let  $F\vec{\delta} = \vec{c}$  be the set of loop basis equations (3). Design can then be cast in the form of a constrained minimization problem [20]. To obtain the fastest circuit, it is necessary to minimize  $\Delta t$ , and with a feasible  $\Delta t$  a minimal total designer added delay  $\sum_j \delta_j$  is desirable. If  $w$  is an arbitrary weight, a standard linear programming problem formulation is

$$\text{minimize } w \cdot \Delta t + \sum_j \delta_j \text{ subject to } F\vec{\delta} = \vec{c} \text{ and } \vec{\delta} \geq 0. \quad (5)$$

Formal methods for solving the problem include the simplex method [21] and the shortest path method [22]. Both methods have complexity of order  $n |E|$  where  $n$  is the number of vertices and  $|E|$  the number of edges in the graph, [20]. Since component fan-in and fan-out is limited,  $|E| = n$  and computational complexity is order  $n^2$ . The above formulation shows that the clock period  $\Delta t$  depends precisely on the circuit structure and device latencies.

### 2.2.2 Bandwidth Considerations

Bandwidth limitations arise from both device switching bandwidth and delay uncertainty. The switching bandwidth of the slowest device in the system must be large enough to allow a clock period of  $\Delta t$ . If delay uncertainty is small, as it is in the discrete component optical circuits built so far,  $\Delta t$  can be determined from the latency equations, and afterwards the amount of stretching required in the clock gates of Fig. 3 can be determined from estimates of the uncertainty. This is discussed in further detail in [20]. Bandwidth can be decoupled from latency by time multiplexing  $m$  copies of the same machine on the hardware. In this mode,  $m$  different pulse trains  $i$  for different values of  $i \bmod m$  form  $m$  independent sequential machines. If the system clock period is  $\Delta \tau$ , each multiplexed machine has an effective period of  $\Delta t = m \cdot \Delta \tau$ . Device switching bandwidth limits take the simple form

$$m \leq \Delta t \cdot \min_{c \in S} \omega(c), \quad (6)$$

where  $\omega(c)$  is the bandwidth of device  $c$  and  $\Delta t$  is determined by the latency equations (3). Delay uncertainty limitations on a multiplexed circuit come from the need to stretch the clock gating signal sufficiently to cover absolute timing uncertainty within the constraints of the smaller clock period  $\Delta\tau$ . Time multiplexing is discussed further in [23].

### 3. Experimental Work

#### 3.1 Implementation Domain

It was our intent from the beginning of our efforts to not only design, but also implement a stored program digital optical computer. We felt that construction and operation of a proof-of-principle machine would go much further than a pencil and paper design to both establish feasibility of the technique, and to drive optical and optoelectronic component technology. Selection of optical and optoelectronic components was very limited. After careful consideration of available components, we settled on  $LiNbO_3$  directional couplers as switches, 3dB or other ratio optical couplers for signal fan-out and fan-in, and optical fiber for interconnections. Switches were available to us from AT&T as experimental products, [24] and consisted of six, 2x2 bypass exchange switches in a single package. The switches, couplers, and fiber delay elements are depicted in Fig. 5. As received, each switch has two optical inputs, A and B, two optical outputs, D and E, and an electronic control, C. The switches employ an approximately five volt switching signal at terminal C to switch them from the “cross”, or exchange state, to the “bar”, or bypass state. They also require static bias voltages in the zero to eight volt range to minimize crosstalk. Since we desired all-optical inputs and outputs, we designed optoelectronic circuitry to convert arriving optical pulses destined for terminal C to electronic pulses, appropriately stretched and thresholded so as to convert the switch to a 5-terminal optical device. The optoelectronic circuitry for terminal C conversion consisted of a GaAs PINFET receiver, 1 GHz bandwidth amplifier, four ECL NOR gates for pulse stretching, a GaAs multiplexer for electronic signal input, and buffers to drive the switch. The electronic signal input was used to load the machine main memory. The pulse stretching done electronically in the prototype can be done by merging two time shifted optical signals, but electronic stretching was more cost effective. More details are given in [19].

The AT&T  $LiNbO_3$  switches used were polarization dependent, thus the inputs required in-line polarization controllers or “butterflies” [25]. These manually operated butterflies modify the polarization state to that required for the switch input. To reduce latency the butterflies were reduced to their minimum possible size [26]. The final switch specifications were as shown in Table 1. Each six-pack switch was packaged into a module containing drive electronics, butterfly polarization rotators, and connectors. Each module was enclosed in a 10x30x40cm chassis.

The switch is a logically complete element and can perform active OR, but 3dB couplers are frequently used as wired Ors to reduce cost. Because of the high cost of the switches and associated drive electronics, we chose a bit-serial implementation. This reduces the number of switches in the data path by a factor of the word length. The essential control, storage, and arithmetic problems must still be addressed, but the

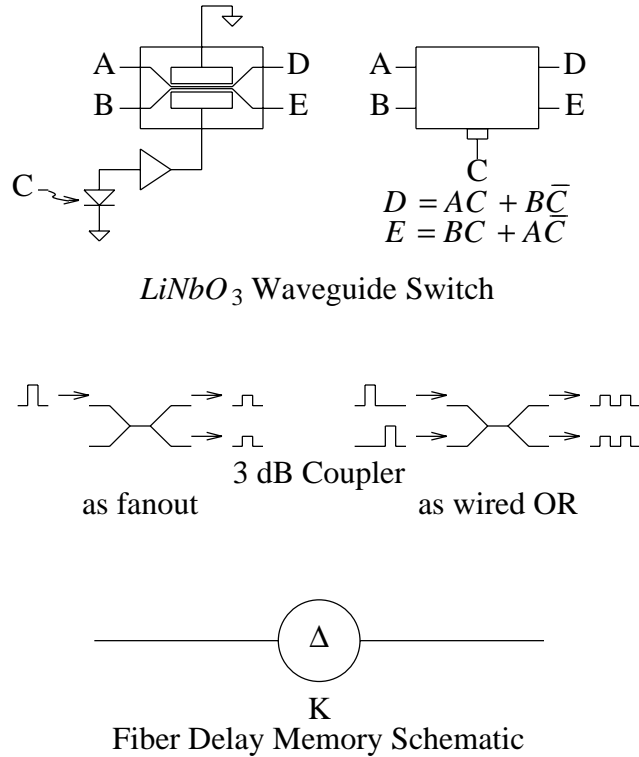


Figure 5: Components used in implementing the digital optical systems

Maximum End-to-End Latency	9ns.
Rise and Fall Times	<2ns.
Insertion Loss	<5dB.
Crosstalk	<20dB.
Sensitivity	$\sim 5 \mu W$
Pulse Stretch	$\sim 4$ ns.

Table 1: Specifications for the electro-optic logic elements

hardware can be minimized.

1310 nm laser diodes were used as signal sources. They were clocked electronically, and clock pulse power was adjusted to  $\sim 1$ mw. The complete computer required about 50mW of optical power.

### 3.2 Signal Quality Restoration

Additional switches were required as clock gates for both timing and amplitude restoration. Insertion loss and switch sensitivity limit to two the number of switches that a signal can traverse before requiring amplitude restoration with clock gating. Estimation of power loss and crosstalk in a complex optical system is itself complex. Details of a graph-theoretic algorithm for computing signal levels is given in [27].

Even if delays can be precisely controlled in optics, timing restoration by clock gating must be applied, at the least, to every feedback loop in the system, because an arbitrarily small timing error can cause a large time shift in a signal stored in a feedback loop for a long period. The problem of determining, for an arbitrary directed graph, a minimum set of edges which, if removed, leave a graph without any directed cycles is the minimum feedback edge set problem [28]. This problem is known to be NP-complete, e.g. [29], but is fairly tractable for real circuits with small fan-out and fan-in.

Finding a minimal feedback edge set involves computing the Boolean permanent (unsigned version of the determinant) of the edge adjacency matrix of the circuit and computing the prime implicants of its dual. If  $f_o$  and  $f_i$  the maximum fan-out and fan-in, respectively of any node, then the complexity order of computation of the permanent is no worse than  $(1 + \min(f_o, f_i))^n$ , where  $n$  is the number of nodes in the circuit graph. Finding prime implicants of a Boolean expression is well studied, and although computationally complex in general, many effective algorithms exist for it.

## 4. Experiments with Optical Time-of-Flight Systems

### 4.1 Binary Counters

The first digital optical circuit built with this method, a serial binary counter, serves as an example of the design method. The lumped delay design of a  $K$  bit counter, which is incremented by a pulse on the  $I$  input synchronous with the low order bit of the count  $s$  is shown in Fig. 6. Components are switched electro-optic directional couplers, fixed 3dB couplers for fan-out as  $f_1$  or fan-in as  $f_3$ , and fibers used for both interconnection and delay elements, as described above. The design implements the circuit equations

$$s(t) = s(t-K\Delta t) \oplus (c(t-\Delta t) + I(t)), \text{ and}$$

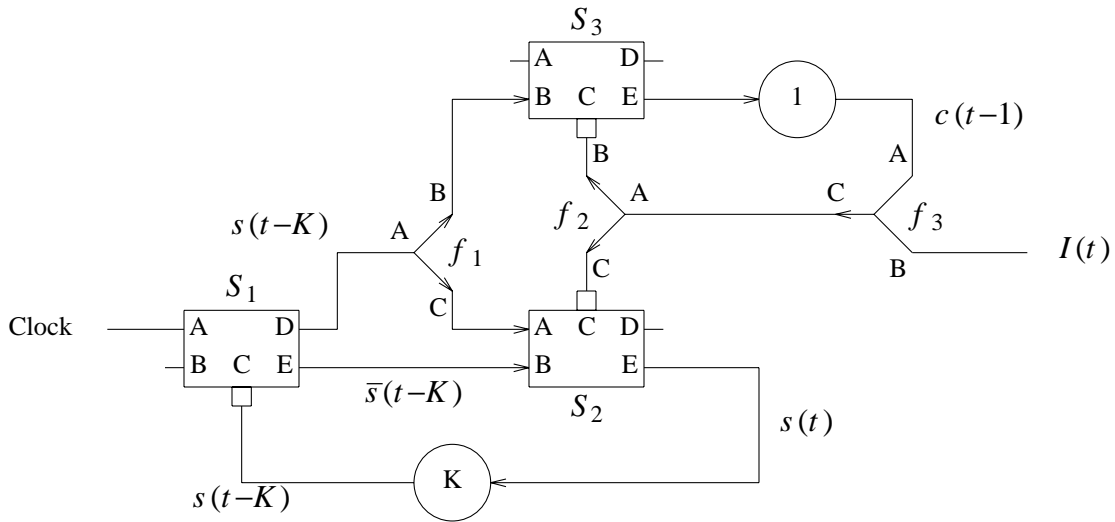


Figure 6: Switched directional coupler circuit for a serial binary counter

$$c(t) = s(t-K\Delta t) \cdot (c(t-\Delta t) + I(t)).$$

The increment input  $I(t)$  is assumed to be zero except for times  $\lfloor t/\Delta t \rfloor = 0 \bmod K$ , with low order bit first serialization. The counter increments if either  $I(t) = 1$  or there was a carry from the high order bit in the previous cycle. The fixed coupler used as fan-in performs a wired OR function.

To illustrate time-of-flight design we form a self contained, free running, counter by deriving the signal  $I$  from the clock with a one-out-of-four scaler  $G_4$  having a known delay from input to output. A graph representation of the free running counter is shown in Fig. 7. Correspondences between the graph nodes and circuit components are given by Table 2. Since explicit delays and the increment generator are two terminal devices, they are not shown as vertices, but are incorporated into edges. The one unit lumped delay is associated with  $e_8$ , the  $K$  unit delay with  $e_2$ , and the increment generator  $G_4$  with  $e_{11}$ . The only nonzero ideal delays are  $b(e_2) = K$  and  $b(e_8) = 1$ . The edge delays for the counter are sums of irreducible component terminal delays  $h$  and designer added delays  $\delta$ . The irreducible delays for edge  $i$  are lumped into a sum  $D_i$  so that  $d(e_i) = D_i + \delta_i$ . For example,  $D_{11} = h(\text{Clock}) + h(G_4) + h(F_{3,B})$ . Thus we can write, for each of the five loops in the basis shown, an equation relating real and lumped delays as in Table 3.

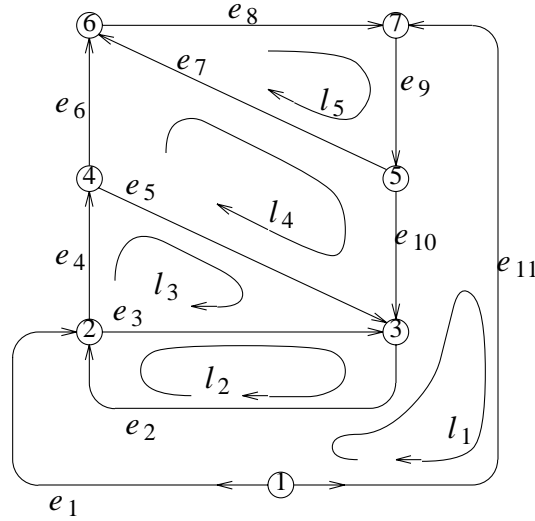


Figure 7: Circuit graph for the serial counter

Node	Circuit component	Node	Circuit component
1	- Clock, with two outputs.	5	- Fixed coupler $f_2$ , for fan-out.
2	- Switch $S_1$ .	6	- Switch $S_3$ .
3	- Switch $S_2$ .	7	- Fixed coupler $f_3$ , for fan-in.
4	- Fixed coupler $f_1$ , for fan-out.		

Table 2: Correspondence between circuit diagram and graph

Loop	Real delay		Lumped delay
$l_1$	$D_1 - D_2 - D_{10} - D_9 - D_{11} + \delta_1 - \delta_2 - \delta_{10} - \delta_9 - \delta_{11}$	=	$-K \Delta t$
$l_2$	$D_3 + D_2 + \delta_3 + \delta_2$	=	$K \Delta t$
$l_3$	$D_4 + D_5 - D_3 + \delta_4 + \delta_5 - \delta_3$	=	0
$l_4$	$D_6 - D_7 + D_{10} - D_5 + \delta_6 - \delta_7 + \delta_{10} - \delta_5$	=	0
$l_5$	$D_8 + D_9 + D_7 + \delta_8 + \delta_9 + \delta_7$	=	$\Delta t$

Table 3: Loop equations from real and lumped delay designs

The problem is to find the smallest  $\Delta t$  such that the equations have solutions for non-negative values of  $\delta_j$ . The solution will not be unique since there will be cases in which reconvergent signal paths allow equal delays to be added to all paths without changing relative timing. The dominant constraint on  $\Delta t$  is often obvious, as in this case, where it is supplied by the one bit feedback loop for the carry. Given a feasible  $\Delta t$ , the constrained minimization objective function can be reduced to  $\sum_j \delta_j$ .

An algorithm to solve the problem of distributing lumped delays over the devices of this circuit is combined with schematic capture and an event driven logic simulator into a computer aided design system called Hatch [30]. Originally written for the Apple Macintosh, it has been transferred to Unix and Xwindows [31] and is now capable of detailed simulations of fairly large circuits. The system not only performs Boolean simulation and delay distribution but also allows specification of optical power loss and crosstalk parameters for devices, using them to identify worst case signal paths and to do a more physical simulation with analog signal levels.

Hatch was used to compute component lengths during the construction of this counter. The Hatch simulation showed that the worst case signal feedback path was through terminals C and E of switch  $S_3$  and the unit delay that forms the carry loop. The minimum delay through this path was 9 ns, thus limiting the clock speed to 100 MHz. Actually, the first counter that we constructed had a different design in which the smallest feedback loop was through two directional couplers rather than one. This limited the clock speed of this first counter to 50 MHz. Details of the 50 MHz counter design and construction are discussed in [17].

The 100 MHz counter described in [19] implemented the design of Fig. 6, and operated at 100 MHz. The same counter was operated as two 50 MHz time-multiplexed counters by doubling the size of the delay loops, thus demonstrating the ability to time multiplex complete systems.

## 4.2 Memory Unit

The next experimental step was the design of a bit serial memory. We constructed a memory consisting of 64 16-bit words. The clock rate was 50 MHz. At this clock rate, one bit length is approximately 4.1 meters, resulting in a total loop length of 4.2 km. End-to-end latency of the loop was thus 20.5  $\mu$ sec. The long memory delay loop made the effects of delay uncertainty evident, but in the simple context of a long recirculating loop. A detailed analysis of physical constraints on the capacity of a synchronous fiber storage loop can be found in [10] and results of the experiments in [32]. The experiments

showed among other things that it was possible to reliably store and retrieve data from the delay line memory. Specifically, only two data bit errors were detected in 154 hours of continuous testing, resulting in a bit error rate of  $6.7 \times 10^{-14}$ .

The complete memory module consists of the delay line memory, a counter signifying the address of the memory location currently arriving at the output of the delay line, and a serial address comparator, as shown in Fig. 8. In operation, the address to be read from or written to (Address in) is repeatedly compared with the address of the word at the output, produced by the Address Counter, and when a match is indicated by the Address Comparator, the word is read or written. Notice that there is no physical connection between the memory loop and the counter and comparator. The connection between them is defined by the *start data transfer* signal, which triggers the the circuitry responsible for memory reads and writes. The labels CLK and WCK respectively represent the master clock and and a word clock having a pulse every 16 bit periods, marking the low order bit of each word.

The bit serial design imposed by high cost components has its most negative impact on performance in the main memory. The time-of-flight design method imposes no constraints on parallel circuitry, but cost dictates careful trade off between the one loop memory described here and the other extreme of one loop per word, or even one per bit.

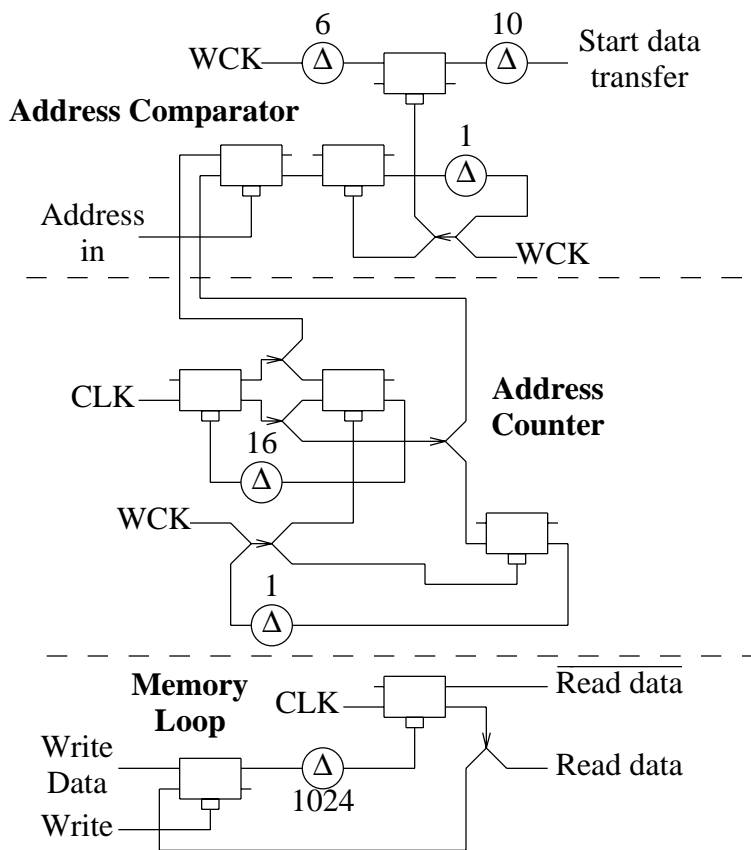


Figure 8: Serial memory of 64 sixteen bit words

### 4.3 The Stored Program Optical Computer

The Stored Program Optical Computer, SPOC, contained an arithmetic and logic unit, a moderate amount of memory, and sufficient control for stored program operation. SPOC showed that a synchronous, time-of-flight, digital optical computer system could be designed [33] and constructed and operated successfully [18]. Operating at 50 MHz., the computer has a simple accumulator architecture with 16 bit words, two's complement arithmetic, a 10 bit memory address, and an instruction set similar to that of a PDP-8 [34]. Only 64 of the possible 1024 words of memory were implemented. The design required 66 electrooptic directional couplers, although 4 were eliminated through some electronic clock generation. The memory used the 4.2 km fiber loop discussed above, and 50 fixed couplers were used for fan-out and, in a few cases, for fan-in. The computer can be seen in Fig. 9. Power supplies are on the bottom shelf, lasers, the memory loop, and clock on the next shelf, and the interconnected six-switch modules on the upper two shelves.

The SPOC proof of principle experiment demonstrated that time-of-flight digital systems are viable in a technology where delays can be precisely controlled. In the process of its development computer aided design algorithms and tools were implemented which make the design of such systems tractable. Although all-optical arithmetic and logic had been previously demonstrated, SPOC provided the first example of substantive stored program control implemented in the optical domain. In the process of debugging the machine, it became evident that with its particular system parameters control of signal delay was not a large problem. More problems were caused by optical power losses and crosstalk in the switches. We are presently engaged in modifying the machine to run two simultaneous time-multiplexed machines each at 50 MHz.

Cost dictated using bulk electrode  $LiNbO_3$  switches with a bandwidth of 200 MHz [35]. The SPOC clock rate of 50 MHz is within a factor of four of this. The 100 MHz clock rate of the time-multiplexed machine will be within a factor of two of the bandwidth, as was the clock rate of the successfully demonstrated 100 MHz counter [19].

## 5. Future Potential of Optical Time-of-Flight Design

This section projects the properties of an optoelectronic integrated circuit (OEIC) version of SPOC. From properties of the discrete component computer and current research in integration technology, parameters such as clock rate, chip area, and power requirements for an OEIC are established. We attempt a balance between the conservative extreme of using only off-the-shelf technology and the optimistic extreme of extrapolating hero experiments into the future.

The technology used to implement SPOC was limited to electrically switched directional couplers, photodetectors, amplifiers, fixed ratio couplers, and waveguides for all storage and interconnection. These can be implemented with a multi-chip approach in which an optical substrate is solder bump bonded to an electronic substrate. The optical substrate supplies integrated directional couplers, both switched and fixed, and waveguides for both interconnection and delay lines short enough to be folded into the available area. The electronic substrate contains photodetectors and amplifiers to drive switch electrodes from a detected optical signal. Coupling of light from the optical into the electronic substrate will be through surface gratings, and the same solder bumps

Figure 9: Photograph of the stored program optical computer

which provide horizontal alignment of the flip chip assembly can connect the driver output to traveling wave switch electrodes. A mode locked laser clock and any long memory delays will be off chip. The scheme is sketched in Fig. 10.

The optical substrate might be  $LiNbO_3$ , some III-V material, or a polymer. Possibilities for the electronic substrate include  $GaAs$ ,  $Si$ ,  $InGaAs$ ,  $InP$ , etc. To make concrete calculations for the integrated system, we assume  $LiNbO_3$  and  $GaAs$ . Experimental data available for these technologies makes prediction possible, even if another material should prove superior in the future. Use of glass fiber for main memory implies an operating wavelength of 1300 nm as in the discrete component version, or perhaps 1540 nm, to satisfy low loss and dispersion over the length of the memory loop. The photodetector technology must also work with 1300 nm. The architectural requirements of SPOC are used for the calculations because it is a general purpose digital system with memory, processing, and control.  $LiNbO_3$  directional couplers have been demonstrated with a switching bandwidth of 40 GHz [7], so we take 20 GHz as an attainable bandwidth for carefully integrated traveling wave switch electrodes and their drivers.

SPOC required about 70 electro-optic directional couplers, about 80 fixed ratio couplers, and 142 bits of delay for working storage and control circuit implementation. The addressable main memory is 1,024 words of 16 bits for a total of 16,384 bits. Integrating the working storage onto the optical substrate in folded waveguide structures leaves only the main memory to be implemented with glass fiber off chip. From these basic assumptions and system requirements, we have estimated the size the of OEIC. If the interconnection and layout overhead is less than .4 cm<sup>2</sup> the entire optical portion of

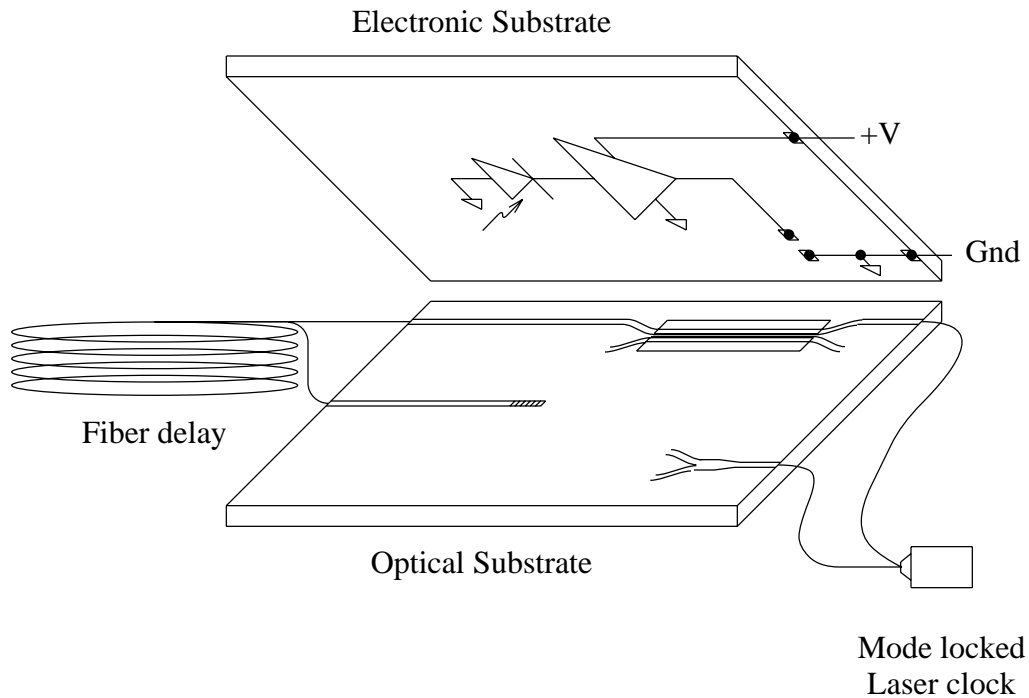


Figure 10: Integrated optoelectronics in a flip chip configuration

the computer, exclusive of main memory, can fit within  $2 \text{ cm}^2$  of  $\text{LiNbO}_3$ . This would still require multiplexing of the system due to the latency limit of one major clock cycle for the counter carry bit feedback loop. Thus for a machine with 500 ps minimum latency, the major clock cycle would be 2 GHz and require a factor of ten multiplexing for the effective machine rate of 20 GHz. The minor clock period would be 50 ps and for the 10 machines to each have 1024 16-bit words of storage would require  $10240 \times 50 \text{ ps} \times 16 \times 2 \times 10^8 \text{ m/s} = 1640 \text{ m}$  memory loop. Note that the long access latency favors a multi-loop memory. Breaking the memory into eight loops would require about 32 more switches [36], which could be supported in the OEIC.

The integration density of the  $\text{GaAs}$  substrate is driven by that of the optical substrate since drivers must be placed corresponding to the switch electrodes they drive. With  $2 \text{ cm}^2$  required by the optical circuit, the density of 70 electronic drivers in a comparable area presents no fabrication problems.

Directional couplers on  $\text{LiNbO}_3$  are well understood, but folding long delay lines onto a small area requires discussion. The two competing technologies for folding are spiral with crossovers and corner mirrors. Bending loss measurements show that spiral waveguides in  $\text{LiNbO}_3$  with a radius of curvature greater than 10 mm have negligible excess loss over straight waveguides [37]. Crossovers to exit from a spiral are not a problem since crosstalk is less than -30 dB in proton exchanged  $\text{LiNbO}_3$  waveguides which cross at an angle greater than 6 degrees [38]. The difficulty of etching  $\text{LiNbO}_3$  is an obstacle to making corner mirrors, but a radius of curvature greater than 10 mm makes S curves practical only for convergence and divergence at the ends of couplers. Laser ablation is a possible way to make corner mirrors in  $\text{LiNbO}_3$ , but it has not been done. Recent work[39] demonstrated ridge waveguides in  $\text{LiNbO}_3$ , and the etching rates achieved could make corner mirrors and sharper bends possible. The low loss of all but very small angle crossovers gives great flexibility in laying out intersecting signals. The grating couplers, e. g. [40], needed to couple light to the detectors have negligible loss, and the dispersion angle is small enough that the  $\sim 30 \mu\text{m}$  spacing of the solder bump bonding allows the detector to collect virtually all of the light.

Sub-micron-gate  $\text{GaAs}$  FET technology can yield high speed optoelectronic receivers [41] as well as high speed amplifiers. The detector favored is a Metal-Semiconductor-Metal detector which can be integrated with different  $\text{GaAs}$  FET technologies, while providing good responsivity, low dark current and high bandwidth [42]. The  $\text{GaAs}$  FET technology can be compatible with  $1.3 \mu\text{m}$  wavelength by adding a  $\text{GaInAs}/\text{GaAs}$  superlattice absorbing region for the MSM detector [43]. Projected receiver sensitivity is better than -20 dBm for 20 GHz at the  $1.3 \mu\text{m}$  wavelength. The MSM detector has a size of  $30 \times 30 \mu\text{m}$  and a responsivity of 0.2 mA/mW. Using a transimpedance receiver circuit for large bandwidth and high sensitivity, four amplifier stages are needed for sufficient current and voltage gain. The switch electrodes act as a  $50 \Omega$  transmission line. Allowing for amplifier power, 40 mW are required for a 1 Volt switch. The solder bumps connecting drivers to electrodes are less than  $30 \mu\text{m}$  in diameter and do not influence the impedance.

The system power is divided into optical power in the  $\text{LiNbO}_3$  chip and fiber and electrical power for the detector, amplifier and driver. In the integrated system, loss per switch is low since the signal seldom passes in and out of fiber. This reduces both optical clock power and the number of switches needed to restore signal level. Not all restoring

switches can be eliminated because any feedback loop must restore both power and timing. Losses in folding long waveguides must also be allowed for. Dropping unneeded restoration switches, no optical path in SPOC passes through more than four 3dB couplers and six switched couplers. With .5 dB loss in an integrated switch and its interconnecting waveguides, such a path would have a loss of 15 dB. We do not count the negligible loss in the grating couplers. Estimating a minimum of -17 dBm at any detector of the OEIC, the maximum power of a gated clock is -2 dBm. Losses due to folding register delays are a few dB, and every register includes regeneration, so these losses are not limiting. Complete estimates of the required optical power range from two to six milliwatts. Optical power is supplied by an off-chip clock generating pulses of about 50% duty cycle at the 20 GHz bit rate. A mode locked semiconductor laser would satisfy this requirement with the worst case power estimate. The electronics associated with each switch requires ~40 mW for a total of 2.8 watts. The power can be easily dissipated with air cooling over the two square centimeter area.

Several research problems are associated with developing such an OEIC. Adapting the general purpose design to a specific application, say for routing and contention resolution in telecommunications, is straightforward. Since SPOC consisted of discrete components connected by fiber, geometric layout was not addressed. An extension to the Xhatch design tool to place and route while maintaining specific path lengths is needed. The design of switches with phase matched electrodes for high speed operation is an important problem. The large number of switches in close proximity and flip chip connection will be new issues. The packing of waveguides has already been mentioned as an incompletely solved problem. Etched, ablated, or possibly distributed Bragg reflection corner mirrors are needed. Design of the surface gratings will concentrate on low loss. Electronic design is dominated by high speed and simplified by the lack of any long signal paths. MSM photodetectors and amplifiers are connected directly to switch electrodes without intervening transmission lines, and the end-to-end latency should be minimized, although it can be compensated for in layout to an extent.

## 6. Conclusions

We have presented time-of-flight design and shown that digital optical circuits can be designed and built in this latch free, dynamic manner. The theory of the design method was presented and illustrated by the design of one of the experimental optical circuits. A series of experiments leading up to the demonstration of a stored program computer showed the scope of the time-of-flight design technique. An important aspect of the method is that it allows clock rates very close to device bandwidth limits. An argument was presented for a general digital OEIC with a 20 GHz bit rate, using optoelectronic integration technology currently available in the research laboratory. The flip chip bonded  $LiNbO_3$  and  $GaAs$  chips are only about two square centimeters in area and dissipate only a few watts of combined optical and electrical power. Although several device integration and systems design problems would need to be addressed to realize such a digital integrated optoelectronic circuit both theory and research experiment results support its feasibility.

## REFERENCES

- [1] J. W. Goodman, "Optics as an interconnect technology," *Optical Processing and Computing*, Chap. 1, Academic Press (1989).
- [2] H. M. Ozaktas and J. W. Goodman, Implications of interconnection theory for optical digital computing," *Applied Optics*, Vol. 31, No. 26, pp. 5559-67 (10 Sept. 1992).
- [3] D. A. B. Miller, "Optics for low-energy communication inside digital processors: quantum detectors, sources, and modulators as efficient impedance converters," *Optics Letters*, Vol. 14, No. 2, pp. 146-148 (15 Jan. 1989).
- [4] M. Haner and W. S. Warren, "Generation of arbitrarily shaped picosecond optical pulses using an integrated electrooptic waveguide modulator," *Applied Optics*, Vol. 26, No. 17, pp. 3687-3694 (Sept. 1987)
- [5] M. N. Islam, C. E. Socolich and D. A. B. Miller, "Low energy ultrafast fiber soliton logic gates," submitted to *Optics Letters*, Vol. 15, No. 16, p. 909 (Aug. 15, 1990).
- [6] K. Uchiyama, H. Takahara, S. Kawanishi, T. Morioka, M. Saruwatari, and T. Kitoh, "100 Gb/s all-optical demultiplexing using nonlinear optical loop mirror with gating-width control," *Electronics Letters*, Vol. 29, No. 21 (14 Oct. 1993).
- [7] S. K. Korotky, et al., "Optical intensity modulation to 40 GHz using a waveguide electrooptic switch," *Applied Physics Letters*, V. 50, pp. 1631-1633 (1987).
- [8] L. M. F. Chirovsky, et al., Field effect transistor — self electrooptic effect device (FET-SEED) circuits for optoelectronic data processing systems," *Proc. Topical Mtg. on Optical Computing*, March 1993, Palm Springs, CA, Optical Society of America 1993 Tech. Digest Series, Vol. 7, pp. 218-221.
- [9] F. B. McCormick, A. L. Lentine, and L. M. F. Chirovsky, "Cascaded operation of two 128-pixel FET-SEED smart pixel arrays," *Proc. Topical Mtg. on Optical Computing*, March 1993, Palm Springs, CA, Optical Society of America 1993 Tech. Digest Series, Vol. 7, pp. 222-225.
- [10] D. B. Sarrazin, H. F. Jordan, and V. P. Heuring, "Fiber optic delay line memory," *Applied Optics*, Vol. 29, No. 5, pp. 627-637 (10 Feb. 1990).
- [11] N. Shenoy, et al., "Retiming of circuits with single phase transparent latches," *28th ACM/IEEE Design Automation Conference*, pp. 549-555 (1992).
- [12] D. C. Wong, G. De Micheli, and M. J. Flynn, "Designing high-performance digital circuits using wave pipelining: Algorithms and practical experiences," *IEEE Trans. on Computer-aided Design of Integrated Circuits and Systems*, Vol. 12, No. 1, pp. 25-46 (Jan. 1993).
- [13] D. A. Joy and M. J. Ciesielski, "Clock period minimization with wave pipelining," *IEEE Trans. on Computer-aided Design of Integrated Circuits and Systems*, Vol. 12, No. 4, pp. 461-472 (April 1993).

- [14] P. J. Delfyett, D. H. Hartman, and S. Z. Ahmad, "Optical clock distribution using a mode-locked semiconductor laser diode system," *J. Lightwave Technology*, Vol. 9, No. 12, p. 1646 (Dec. 1991).
- [15] C. Qiao, R. G. Melhem, D. M. Chiarulli and S. P. Levitan, "Optical multicasting in linear arrays," *Int'nl J. Optical Computing*, Vol. 2, pp. 31-48 (1991).
- [16] A. L. Lentine, D. A. B. Miller, and J. E. Henry, "Optical logic using electrically connected quantum well PIN diode modulators and detectors," *Applied optics*, Vol. 29, No. 14, p. 2153 (10 May 1990).
- [17] A. F. Benner, J. Bowman, T. Erkkila, R. J. Feuerstein, V. P. Heuring, H. F. Jordan, J. R. Sauer and T. Soukup, "Digital Optical Counter Using Directional Coupler Switches," *Applied Optics*, V. 30, No. 29, pp. 4179-4189 (10 Oct. 1991).
- [18] P. T. Main, R. J. Feuerstein, H. F. Jordan, V. P. Heuring, J. R. Feehrer, and C. E. Love, "Implementation of a general purpose stored program digital optical computer," *Applied Optics*, V. 33, No. 8, pp. 1619-1628 (10 March 1994).
- [19] R. J. Feuerstein, T. Soukup and V. P. Heuring, "100-MHz optical counter that uses directional coupler switches," *Optics Letters*, V. 16, No. 20, pp. 1599-1601 (Oct. 15, 1991).
- [20] J. P. Pratt and V. P. Heuring, "Delay synchronization in time-of-flight optical systems," *Applied Optics*, V. 31, No. 14, pp. 2430-2437 (May 10, 1992).
- [21] F. A. Ficken, *The Simplex Method of Linear Programming*, Holt, Rinehart and Winston, New York (1961)
- [22] N. Christofides, *Graph Theory*, Academic Press, London (1975)
- [23] Harry F. Jordan and Vincent P. Heuring, "Time Multiplexed Optical Computers," *Proc. Supercomputing '91*, IEEE Computer Society Press, pp. 370-378 (Nov. 1991).
- [24] F. T. Stone, J. E. Watson, D. T. Moser and W. J. Minford, "Performance and Yield of Pilot-Line Quantities of Lithium Niobate Switches", *Proc. SPIE*, Vol. 1177, pp. 322-325, Integrated Optics and Optoelectronics, at OE Fibers 89, K. K. Wong et al., Eds., SPIE Bellingham, WA (1990).
- [25] B. G. Koehler and J. E. Bowers, "In-line single-mode fiber polarization controllers at 1.55, 1.3, and 0.63  $\mu\text{m}$ ", *Appl. Opt.*, Vol. 24, No. 3 (1 Feb. 1985).
- [26] D. B. Sarrazin, Behavior of Miniaturized Polarization Controllers, OCS Technical Report 89-49 (Dec 1989).
- [27] Jonathan Pratt and Vincent P. Heuring, "Designing digital optical computing systems: power distribution and cross talk," *Applied Optics*, **31**, # 23, 4657-4661 (1992).
- [28] A. Lempel and I. Cederbaum, "Minimum feedback arc and vertex sets of a directed graph," *IEEE Trans. on Circuit Theory*, Vol. CT-13, No. 4, pp. 399-403 (Dec. 1966).

- [29] A. V. Aho, J. E. Hopcroft, and J. D. Ullman, *The Design and Analysis of Computer Algorithms*, Addison-Wesley, Reading, MA (1974).
- [30] J. P. Pratt, "HATCH User's Manual," *OCS Tech. Rept. 89-31*, Center for Optoelectronic Computing Systems, Univ. of Colo., Boulder, CO 80309-0525 (1989).
- [31] M. Salerno, "XHATCH User's Manual," *OCS Tech. Rept. 91-25*, Center for Optoelectronic Computing Systems, Univ. of Colo., Boulder, CO 80309-0525 (October 1991).
- [32] T.J. Soukup, R.J. Feuerstein, and V.P. Heuring, "Implementation of a fiber-optic delay-line memory," *Applied Optics*, V. 31, No. 17, pp. 3233-3240 (10 June 1992).
- [33] V. P. Heuring, H. F. Jordan, and J. P. Pratt, "Bit-serial architecture for optical computing," *Applied Optics*, V. 31, No. 17, pp. 3213-3224 (10 June 1992).
- [34] D. P. Siewiorek, C. G. Bell, and A. Newell, *Computer Structures: Principles and Examples*, Chap. 8, McGraw-Hill, New York (1982).
- [35] R. J. Feuerstein, "Lithium niobate switch evaluation," *OCS Tech. Rept. 90-16A*, Center for Optoelectronic Computing Systems, Univ. of Colo., Boulder, CO 80309-0525 (May 1990).
- [36] David B. Sarrazin, "Optical delay line memory systems," Ph.D. Thesis, *OCS Tech. Rept. 92-18*, Center for Optoelectronic Computing Systems, Univ. of Colo., Boulder, CO 80309-0525 (April 1992).
- [37] I. Januar and A. Mickelson, "Low-loss PE:LiNbO<sub>3</sub> waveguide bends at  $\lambda = 1.55 \mu\text{m}$ ," *Proc. Integrated Photonics Research Conf.*, New Orleans (April 1992).
- [38] I. Januar and A. R. Mickelson, "Characteristics of S-shape waveguide structures by annealed proton exchange process in LiNbO<sub>3</sub>," accepted for publication in *J. Lightwave Technol.*, (1993).
- [39] K. Noguchi, O. Mitomi, and K. Kawano, "Highly efficient 40-GHz Recent work[39] demonstrated ridge waveguides in LiNbO<sub>3</sub>, and *Photonics Tech. Lett.*, Vol. 5, No. 1, pp. 52-54 (1993).
- [40] Q. Huang and H. J. Caulfield, "Waveguide holography and its applications," *Proc. SPIE*, V. 1461, Practical Holography V, SPIE Press, Bellingham, WA (1991).
- [41] A. Ketterson, J. W. Seo, M. Tong, K. Nummila, D. Ballegeer, S. M. Kang, K. Y. Cheng, and I. Adesida, "A 10 GHz bandwidth pseudomorphic GaAs/InGaAs/AlGaAs MODFET-based OEIC receiver," *Proc. Device Research Conf.*, Massachusetts, VIB-5 (1992).
- [42] B. J. Van Zeghbroeck, W. Patrick, J.-M. Halbout, and P. Vettiger, "105 GHz bandwidth metal-semiconductor-metal photodiode," *IEEE Electron Dev. Lett.*, V. EDL-10, pp. 527-529 (1988).
- [43] M. Zirngibl and M. Ilegems, "High-speed photodetectors on InGaAs/GaAs-on-GaAs superlattices," *J. Applied Physics*, V. 69, pp. 8392-8398 (1991).

