Systems considerations in designing and implementing a bit serial optical computer

Vincent P. Heuring, Member SPIE
University of Colorado
Department of Electrical and Computer Engineering
and
Center for Optoelectronic Computing Systems
Boulder, Colorado 80309-0425

Abstract. The systems considerations in implementing a bit serial optical computer using lithium niobate directional couplers and optical fiber is described. The three main problems in constructing such a computer, computation of fiber lengths, estimation of and compensation for power losses, and details of the physical construction of the machine, are discussed.

Subject terms: optical computing; fiber optics; lithium niobate; directional couplers; computer architecture.


CONTENTS
1. Introduction
2. Design of the SCAMP
3. The effect of loss and crosstalk in lithium niobate switches
4. The effect of finite component length on system design
5. Summary of the design process
6. Construction details
   6.1. The switch module
   6.2. Calibration and testing
7. Future plans
8. Acknowledgment
9. References

1. INTRODUCTION

We are in the process of constructing a bit serial optical computer using lithium niobate directional couplers as logic elements and optical fibers for interconnection. This paper describes the systems issues that are being addressed in the design and construction of such a machine. These issues fall into three main areas:

- The need for an algorithm to compute fiber lengths;
- The need to incorporate amplitude restoration and signal resynchronization; and
- The design and construction of a practical switch module incorporating switch, connectors, polarization controllers, and drive electronics.

After a brief description of the machine architecture, the remainder of the paper discusses these three topics.

2. DESIGN OF THE SCAMP

In the original design of the bit serial computer, it was assumed that the LiNbO₃ switches were ideal components: the components had no loss, no crosstalk, and no physical length. This is the most practical approach to initial system design, since it permits the architect to concentrate on the system logic and architecture, without being concerned about the physical limitations of the components being used. Nonidealities in the components are then considered in a later phase of the design process.

The SCAMP design is a single address machine, with accumulator, program counter, instruction register, arithmetic and logic unit (ALU), and main memory. Figure 1 shows the data paths in the SCAMP design. Registers and main memory are implemented using recirculating optical fiber delay loops rather than bistable elements such as flip-flops. The figure shows the memory counter (MC) that is used to synchronize memory references. The design relies on carefully trimmed fibers rather than flip flops for signal synchronization.

Figure 2 shows the Boolean logic functionality of the LiNbO₃ switch, where terminal C contains an optical to electronic interface, making the switch a five-terminal optical device. When terminal C is high, signals entering at terminal A are switched to terminal D, and signals entering at terminal B are switched to terminal E. This is referred to as the "bar" state. When terminal C is low, signals entering at terminal A are switched to terminal E, and signals entering at terminal B are switched to terminal D. This is referred to as the "cross" state.

Figure 3 shows how a recirculating fiber delay line memory is implemented using LiNbO₃ switches as logic elements and optical fibers as delay elements. Notice that although information propagates around the loop continuously, the optical pulses travel around the loop only once, terminating at terminal C of switch 1, where they function to switch a fresh copy of the clock into the loop. This has the salutary effect of restoring the amplitude of the data pulse, and also resynchronizing the incoming signal to the arriving clock signal. Figure 4 shows how resynchronization is accomplished by stretching the signal arriving at ter-

---

Paper 2949 received July 17, 1990; revised manuscript received June 28, 1991; accepted for publication July 5, 1991.
© 1991 Society of Photo-Optical Instrumentation Engineers.
Fig. 2. Functionality of the lithium niobate switch.

Fig. 3. Fiber-optic delay line memory.

Fig. 4. Use of a lithium niobate switch to restore signal amplitude.

A terminal C by an amount sufficient to mask any jitter in the signal arriving at terminal A. Such jitter may be due to thermally induced refractive index changes, slight errors in fiber lengths, or signal amplitude degradation. Details of the design and construction of fiber delay loops are discussed by Sarrazin.\footnote{The Hatch software and manual is available on request to the Center at the above address, as Technical Report # 89-31.}

3. THE EFFECT OF LOSS AND CROSSTALK IN LITHIUM NIOBATE SWITCHES

The switches are custom devices, manufactured by AT&T Bell Labs,\footnote{The Hatch software and manual is available on request to the Center at the above address, as Technical Report # 89-31.} and are packaged six to a package. They operate at 1300 nm, switch at <5 ns, have less than –20-dB crosstalk isolation, less than 5-dB insertion loss, and are polarization dependent.

The sensitivity of the drive electronics at terminal C is normally adjusted to –17 dBm; since signals suffer up to 5-dB loss when passing through a switch, a 1-mW data signal can travel through only three switches before its amplitude has diminished to an extent that it will not actuate a switch. The signal restoration scheme described above is used to restore the signal to full amplitude, as shown in Fig. 4; a weak signal arriving at terminal C switches a fresh copy of the clock pulse into the output.

The design also uses conventional 3-dB optical splitters for signal fan-out, and optical combiners that serve as the optical equivalent of the “wired or” gate for fan-in; thus signals suffer a 3-dB loss when passing through these splitters and combiners.

We have developed a graph theoretic model that allows us to analyze these “ideal” architectures and to compute minimum power levels and maximum crosstalk at all points in the circuit.\footnote{The Hatch software and manual is available on request to the Center at the above address, as Technical Report # 89-31.}

The graph model employs a directed graph whose root is the power source of the optical system and whose leaves are the detection (and termination) points of the optical signals—typically terminal C of the LiNbO$_3$ switches. Each device in the system is modeled as a set of vertices at the inputs coupled to a set of vertices at the outputs. The device is modeled as having multiple independent states in which the coupling terms between the inputs and outputs vary. The input-output coupling is represented by weights applied to directed edges between each input and each output. There are two types of coupling: noise and signal. Noise coupling arises because of crosstalk from nonselected inputs to a given output, and signal coupling is the desired logical path of an optical signal from input to output. A depth-first search of the graph is made, tracking the values of maximum power, minimum power of logical 1s, and maximum power of logical 0s, from the root along all possible paths to every signal detection point. The search algorithm detects when power at a detection point is inadequate to switch the device, and/or when crosstalk results in incorrect device operation.

This model has been incorporated into our graphical design and simulation tool, Hatch,* developed for the Apple Macintosh computer.\footnote{The Hatch software and manual is available on request to the Center at the above address, as Technical Report # 89-31.} Figure 5 shows a typical Hatch screen. Note the ability to select components from a tool menu. This menu includes all the components needed for the design of serial systems using directional couplers. Various components, such as clock

---

*The Hatch software and manual is available on request to the Center at the above address, as Technical Report # 89-31.
sources, switches, splitters, fibers, detectors, etc., can be selected and "fibered" together to form a system. Hatch will then simulate the logical operation of the system, as shown in Fig. 6. After providing Hatch with the loss and crosstalk parameters of all devices, the power analysis algorithm described above pinpoint all circuit nodes where power loss or crosstalk is too great for the circuit to operate correctly. We used this tool to analyze the original SCAMP design, adding power restoration wherever appropriate. The heuristic developed for adding the minimum number of restoration nodes is interesting in itself, and will be described in a future publication. The drive to minimize the total number of switches is powerful, since the present cost per switch is $2000, not counting drive electronics and connectors. The original SCAMP design used 48 LiNbO₃ switches; this number increased to 72 switches when signal loss and restoration was incorporated into the design.

4. THE EFFECT OF FINITE COMPONENT LENGTH ON SYSTEM DESIGN

If components had zero physical length, then components with no desired fiber delay between them could be placed arbitrarily close, and desired fiber delays could be computed from the velocity of light, the optical pulse length, and the refractive index of the medium, from

$$l = \frac{nc}{v\tau}, \quad (1)$$

where \(l\) is the fiber length, \(n\) is the number of clock periods of desired delay, \(c\) is the velocity of light, \(v\) is the clock frequency, and \(\tau\) is the refractive index of the fiber.

Of course, components do have finite lengths, and these lengths must be accounted for by distributing them into and subtracting them from the lengths of fibers where delays are desired. As in the case of power distribution discussed above, we have developed an algorithm for automatically distributing component delays into lumped delays. The method converts the design into a directed graph in which signal interaction points are nodes and the directed signal paths are edges. The resulting algorithm will distribute delays into available delay loops if possible, and will fail if such a distribution is not possible. This delay distribution algorithm has also been incorporated into Hatch, and is used to compute actual fiber lengths to be used, once component lengths are known.

5. SUMMARY OF THE DESIGN PROCESS

The design process proceeds as follows. Construct an initial architecture that employs LiNbO₃ switches as logic elements, and uses lumped delays where they are needed as synchronizing and memory elements. After the lumped delay design has been verified, the effects of power loss and crosstalk are introduced, and compensated for by adding switches where needed to restore signal amplitude and synchronization.

Following this, the fiber lengths are calculated by distributing component lengths into those parts of the design where there are lumped delays. It is this process that introduces an upper bound on the system clock: notice that Eq. (1) relates the clock period to the length of a fiber required to produce a one-clock delay. Roughly speaking, the upper bound on clock frequency is that point where the physical length of a component becomes an appreciable fraction of the length of the smallest fiber delay loop. This is normally the length of a fiber representing one bit delay. And as discussed below, the component "length" that is the longest is the "length" of terminal C. This "length" is the physical length of the fiber and interconnections, plus the latency of the drive electronics. The latency is <10 ns in our system.

6. CONSTRUCTION DETAILS

6.1. The switch module

Figure 7 shows a photograph of a switch module incorporating six LiNbO₃ switches, complete with connectors, detectors and drive electronics, and polarization controllers. The polarization controllers are needed because the switches require polarized light, and we do not use polarization-preserving fiber. The use of short lengths of non-polarization-preserving fiber introduces a small degree of elliptical polarization, which is removed by the polarization controllers. The design and operation of these controllers is discussed by Sarrazin et al. The details of the construction of the optical detector and drive electronics are discussed by Benner et al.

6.2. Calibration and testing

This section discusses the measurement of fiber refractive index and determination of terminal C delay. The terminal lengths of terminals A, B, D, and E are determined by direct physical measurement. The delay measurement at terminal C poses more of a problem, since its delay is the sum of the physical lengths of the components and the latency in the drive electronics. The experimental setup in Fig. 8(a) permits the measurement of both fiber refractive index and terminal C delay. The figure shows a cw laser input to a switch, configured as an oscillator, with a feedback path from terminal E to terminal C. The intrinsic delay in the feedback loop is shown symbolically as a circled A. The circuit operates as follows. Initially the switch is dark, and thus in the cross state. In this state, the first light from the cw laser enters at terminal A and emerges at terminal E. When the light has traveled all the way around to terminal C, the switch switches to the bar state, light entering at terminal A emerges at terminal D, and the loop empties. Once the loop is dark, the switch returns to the cross state, and the process repeats. The period of oscillation \(T_{osc}\) is given by \(2(\tau_E + \tau_C + \tau_f)\), where \(\tau_E\) and \(\tau_C\) are the propagation times through terminals E and C, respec-

Fig. 7. The switch module.
Fig. 8 (a) Circuit for measuring terminal C delay and fiber refractive index; (b) operation of the circuit in (a); (c) period of oscillation of the circuit in (a) as a function of fiber loop length.

\[ \tau_{\text{tot}} = 2\tau_{\text{dev}} + \frac{2l}{c} \]  

where \( \tau_{\text{dev}} \) is the total delay through terminals E and C; and \( l \), \( \eta \), and \( c \) are as defined previously. The oscillatory behavior of this circuit is shown in Fig. 8(b). In order to measure refractive index and \( \tau_{\text{dev}} \), we plot oscillation period as a function of added fiber length \( l \). Figure 8(c) shows such a plot. Data extracted from this plot show a total device delay \( \tau_{\text{dev}} \) of 9.29 ns and a fiber refractive index of 1.473.

As a test of these data, the same experimental setup was used, except the fiber was trimmed to make the total loop delay \( 2\Delta \), where \( \Delta \) is the period of a 40-MHz clock, and the cw laser source was modulated with a 40-MHz square wave, as shown in Fig. 9(a). The behavior of this experiment should be that two pulses enter the loop, and since they arrive at the switch just in time to switch the next two pulses out of the loop, the next two pulses do not enter the loop, and the process repeats. The operation of this “two-up-two-down” oscillator is shown in Fig. 9(b). The figure also shows the system clock.

7. FUTURE PLANS

Further experiments with pulse variability have shown that it would be advantageous to have the pulse stretched at terminal C an additional amount, probably 10 to 20%. Therefore we are redesigning the drive electronics to give us better control over the amount of pulse stretching. Following this, we intend to build a counter and memory loop, as a prelude to the construction of the complete bit serial optical computer.

8. ACKNOWLEDGMENT

The Center for Optoelectronic Computing Systems is sponsored in part by NSF grant number CDR 8622236 as part of the Engineering Research Centers Program, and the Colorado Advanced Technology Institute (CATI), an agency of the State of Colorado.

9. REFERENCES


Vincent P. Heuring is assistant professor of electrical and computer engineering at the University of Colorado, Boulder. He is presently program manager of the Digital Optical Computing Program within the Center for Optoelectronic Computing Systems at the University of Colorado. He received his BS degree from the University of Cincinnati, and his Ph.D. degree from the University of Florida at Gainesville. His major research interests are optical computing, computer architecture, and programming languages.