SF 298

REPORT DOCUMENTATION PAGE

Form Approved
OMB NO. 0704-0188

1. AGENCY USE ONLY (*Leave blank*)

2. REPORT DATE
March 29, 1996

3. REPORT TYPE AND DATES COVERED
Interim Progress Report, 1 Jun 1995–31 Dec 1995

4. TITLE AND SUBTITLE
Smart Photonic Networks and Computer Security for Image Data

5. FUNDING NUMBERS
DAAH04-95-1-0123

6. AUTHOR(S)
M. J. Flynn, M. Morf, J. Gill

7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES)
Stanford University
Sponsored Projects Office
857 Serra Street, Room 260
Stanford, CA 94305-4125

8. PERFORMING ORGANIZATION REPORT NUMBER

9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES)
U.S. Army Research Office
P.O. Box 12211
Research Triangle Park, NC 27709-2211

10. SPONSORING/MONITORING AGENCY REPORT NUMBER

11. SUPPLEMENTARY NOTES
The views, opinions, and/or findings contained in this report are those of the author(s) and should not be construed as an official Department of the Army position, policy, or decision, unless so designated by other documentation.

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

12b. DISTRIBUTION CODE

13. ABSTRACT (*Maximum 200 words*)

The initial focus was on three major areas: Coding and Security, Switching Architecture Simulations, and Basic Technologies, including their interactions. Our new space-time code diversity based algorithms for image and other types of data provide integrated security (privacy, integrity, reliability, and availability) by exploiting parallelism and scalable multiplexing schemes to build photonic network architectures. A number of very high-speed switching and routing architectures and their relationships with very high performance processor architectures and the necessary simulation tools were studied. Routers for very high speed photonic networks can be designed using the very robust and distributed TCP/IP protocol, if suitable processor architecture support is available. Lower levels require very fine grained, stream oriented micro-architectures that encompass a proper combination of current and future CMOS technology and emerging photonic technologies (fiber, photonic transmission line, and 3-D quantum devices). Our basic technology studies included fundamental limits and a number of photonic devices. 3-D quantum devices under development at Stanford (e.g., based on quantum dots) can be significantly smaller than the wavelength of light. To maximize the utility of these quantum dots, super-resolution of light wave detection is required. Our studies suggest that this is at least conceptually feasible.

14. SUBJECT TERMS
Integrated data security, space-time code diversity, network architecture and processing, bandwidth and latency of networks and switching devices, VLIW network processors, X-modulator and X-gate devices, NV simulator, fundamental limits of optical switching and memory.

15. NUMBER OF PAGES
24

16. PRICE CODE

17. SECURITY CLASSIFICATION OF REPORT
UNCLASSIFIED

18. SECURITY CLASSIFICATION OF THIS PAGE
UNCLASSIFIED

19. SECURITY CLASSIFICATION OF ABSTRACT
UNCLASSIFIED

20. LIMITATION OF ABSTRACT
UL

Contents

1 List of Manuscripts Submitted or Published

1. M. J. Flynn. *Computer Architecture: Pipelined and Parallel Processor Design.* Jones & Bartlett, 1995. 788 pages.

2. David B. Glasco. Design and Analysis of Update-Based Cache, Coherence Protocols for Scalable Shared-Memory Multiprocessors. CSL-TR-95-670, June 1995.

3. J.S. Powell, J.A. Trezza, M. Morf, and J.S. Harris, Jr. Vertical Cavity X-Modulators for WDM. In *MPPOI'95*, October 1995.

4. Daniel F. Zucker, Michael J. Flynn, and Ruby B. Lee. A Comparison of Hardware Prefetching Techniques for Multimedia Benchmarks. CSL-TR-95-683, December 1995.

5. J.A. Trezza, M. Morf, and J.S. Harris, Jr. Creation and Optimization of Vertical Cavity X-Modulators. *IEEE Journal of Quantum Electronics* 32(1):53–60, January 1996.

6. James E. Bennett and Michael J. Flynn. Latency Tolerance for Dynamic Processors. CSL-TR-96-687, January 1996.

7. D. Zucker, M. J. Flynn and R. B. Lee. Improving Cache Memory Performance for MPEG Players. In *Proceedings of COMPCON'96*, IEEE-CS Press. February 1996.

8. M. J. Flynn and K. W. Rudd. Parallel Architectures. *ACM Computing Surveys*, 50th Anniversary of Electronic Computing Special Issue. In press, 1996.

9. M. J. Flynn and K. W. Rudd. Parallel Architectures. Chapter in Encyclopedia of Computer Architecture. Alan Tucker, ed., CRC Press (in press) 1996.

2 Scientific Personnel Supported and Honors/Awards/Degrees

Professor Michael Flynn (Sections 4.2, 4.4)

- Received the Harry J. Goode Memorial Award, in recognition of an outstanding contribution to the information processing field "for pivotal seminal contributions to the design and classification of computer architecture." IEEE, 1995.

Professor John Gill (Sections 4.1, 4.2)

Visiting Professor Martin Morf (Sections 4.1, 4.2, 4.3, 4.5)

Dr. Frederick Blau (Sections 4.5)

Dr. Alan Huang (Sections 4.2, 4.3)

Kevin Rudd (Section 4.4)

Jorge Campello (Sections 4.1, 4.2)

Karin Wells

PhD (*Integrated Optical Interconnects*), Electrical Engineering Department.
Expected Sept. 1996.

3 Report of Inventions

1 patent filed, preliminary work on 3 patent disclosures.

4 Scientific Progress and Accomplishments

The objective of this project is the investigation of technologies for the design of very high speed networks for the communication and distributed processing of image-type data. We study new coding and routing algorithms together with network architectures that are matched to optical fibers and photonic switching devices. To achieve these goals, at least three major areas need to be addressed.

- Coding for Data Security

  The goal is to provide secure high-speed data communications for capturing, processing, and transmitting image-type data. These tasks impose major challenges for the design of secure information networks/processing systems. To provide the fundamental security needs for high bandwidth optical communication links, we have investigated an approach that achieves privacy, integrity, availability, and reliability using a integrated, unified coding scheme of low complexity that can be implemented using photonic technologies.

- Network Architecture and Processing

  We have considered very high-speed switching technologies, various routing architectures and their relationships with very high performance processor architectures, including necessary simulation tools. A number of candidate protocols such as TCP/IP were studied, focusing on those that satisfy the robustness and distributed processing requirements of DOD applications. To support such protocols, we studied suitable processor architectures that can handle both highly parallel fine-grain tasks found in stream oriented communications problems, as well as more complex but more infrequently executed code, such as high level communications protocols and applications.

- Basic Technologies.

  Construction of very high speed photonic networks requires access to basic technologies for network links and nodes. We have studied a number of device technologies, their physical limits, and their impact on network parameters such as bandwidth and latency. At the device level, 3-D quantum devices are being studied at Stanford and elsewhere. Our studies included investigations of physical limits entering device level designs. In particular, quantum well devices (e.g., based on quantum dots) can be significantly smaller than the wavelength of light. To maximize the utility of these quantum dots, *super-resolution* light wave detection is required. In an ongoing study, we have determined that such super-resolution is conceptually feasible.

These three problem areas will be addressed by Smart Photonic Interconnect Networks (SPINs) for secure and robust high speed image data communications. Given user and mission requirements, a flexible architecture with intelligent and reconfigurable subsystems is required. The challenge is to find an appropriate network architecture that meets these requirements by careful design of the hardware and software network technologies, while satisfying other constraints such as cost, scalability, and organizational requirements.

4.1 Coding for Data Security

The goal of this project is to design Smart Photonic Interconnect Networks (SPINs) that provide secure and robust high speed data communication needed for capturing, processing, and transmitting image-type data. These tasks impose major challenges for the design of reliable and secure information networks and processing systems. Designing optimal SPINs raises a number of issues, including coding techniques to provide for security needs, such as privacy, integrity, availability, and reliability of high bandwidth optical communication links.

To meet the requirement for reliability and robustness against link failures in networks, several versions of each message must be sent over different paths and/or at different times. However, this increased traffic increases the probability of interception. We have studied a number of techniques that are candidates to deal with security aspects of systems architectures. One of the more promising techniques is the use of *secret sharing*. We consider this method as a special case of a more general concept of *space-time code diversity*. Instead of simply sending multiple copies of the message, *shares* of the secret message are transmitted over different paths. The message shares are generated from the original message in such a way that the message can be reconstructed from a specified minimum number of shares, but no information is gained from a smaller number of shares.

4.1.1 Integrated Data Security

The very high speed optical communication characteristics of SPINs present several challenges in the area of data security. Challenges arise from the high data rates and the corresponding large latency (round trip delay measured in bits). The coding and encryption methods that were designed to perform at speeds appropriate for electronic communications may be too slow for optical data rates. Simple protocols such as ARQ (Automatic Repeat reQuest) require extremely large buffers to store real-time data for possible retransmission.

The primary security needs for SPINs are:

Privacy: The confidentiality of the data in the network should be maintained even under the assumption that an eavesdropper can tap several links in the network.

Integrity: It must be assured that the data is not modified accidentally or deliberately in transit, by replacement, insertion, or deletion.

Reliability: The data communication should be robust enough to withstand link failures and misrouting.

Availability: The communication network should be flexible enough to maintain a minimum level of throughput even under the severe constraints enforced by high rate image data and link failures.

We have investigated integrated approaches that solve these data protection issues using unified coding methods. The several requirements impose different types of constraints that are usually not satisfied by traditional coding schemes.

As pointed out above, there is an important tradeoff between reliability and privacy. The classical approach to this problem is to first encrypt the data to obtain the desired level of privacy and then use error control codes to ensure reliability. This simple approach cannot be applied to the scenario envisioned here, since existing encryption techniques cannot provide the necessary privacy levels at the very high data rates required by image data in all-optical networks.

To achieve both privacy and reliability, we use *secret sharing* techniques, which can be considered to be a form of space-time code diversity. Instead of simply sending multiple copies of the message, shares of the secret message are transmitted over different paths.

A (k, n) secret sharing system breaks the secret s from a set S into n pieces, $v_1, v_2, \ldots, v_n$, each chosen from a set V, such that the following conditions are satisfied:

1. The secret s is recoverable from any k pieces ($k \leq n$).

2. Knowledge of $k - 1$ or fewer pieces provides absolutely no information about s.

To avoid excessive data expansion, we also require:

3. $|V| \leq |S|$. That is, each piece v_i is no longer than s.

We are generally interested in the case where s is chosen uniformly from S. In this case, $|V| = |S|$ in order to satisfy condition 2. Hence, if the number of distinct paths that reliably transmit the message is greater than the maximum number of paths an eavesdropper can tap, the concept of secret sharing can be used to provide both reliability and privacy.

Integrity can also be obtained by the use of secret sharing. To obtain integrity, the receiver waits until more than the minimum number of shares have arrived and then verifies that several subsets of the shares reconstruct the same message. Thus an intruder would have to modify a large number of shares in order to successfully modify the contents of a message without being detected.

Availability is obtained when secret sharing is combined with routing and priority schemes. The idea is to send more shares than needed to provide the desired levels of privacy, reliability, and integrity and then have the network decide locally on when a certain share should be dropped to resolve possible collisions of shares for different messages. This provides the necessary flexibility to cope with stringent availability constraints of highly variable demand high rate image data on optical networks.

In spite of the conceptual suitability of secret sharing to fulfill the security needs in SPINs, it is important to note that secret sharing was designed with another purpose in mind. Traditionally, secret sharing techniques aimed at providing reliability and privacy for a small amount of highly sensitive information, such as keys for private-key cryptosystems. Local memory efficiency and recovery complexity were generally not an issue. Existing secret sharing methods, such as polynomial interpolation schemes, are not suitable for high speed optical environments. We have explored low complexity encoding schemes that can be implemented in optical technologies under consideration.

One approach is to use the secret sharing scheme presented in [Karnin, Greene, Hellman, 1983]. To send a message $m \in GF(q)$ one forms a vector of length k over $GF(q)$ whose first component is m and whose remaining $k-1$ components are chosen uniformly at random from $GF(q)$. Then this vector is multiplied by a $k \times n$ matrix A to obtain a vector of length n over $GF(q)$ whose components are the shares. The matrix A should have the property that any $k \times k$ submatrix is nonsingular. If the matrix A is chosen at random, then with reasonably high probability it has the desired property. However, in practice, random choice of the matrix A does not lead to computationally efficient or even feasible reconstruction procedures. Therefore the matrix A should be structured enough to allow for computationally efficient algorithms. We have investigated the use of parity-check matrices for Reed-Solomon codes. This leads to a potential for taking advantage of X-gates via their associated group codes.

4.1.2 Space-Time Code Diversity and Implementation Issues

As mentioned above, there are a number of new aspects to be considered when applying secret sharing techniques to provide the security needs of SPINs. For absolutely reliable communications, it is necessary that data be stored at a network node until the data is known to have been transferred successfully to another node, which is usually closer to the ultimate destination. Therefore the network entry nodes to which data sources directly connect must provide storage for sensors without memory.

The high data rates associated with data fusion networks contribute directly to an increase in the communication latency measured in bits. This high latency conflicts with another requirement, that data sources are assumed to be simple sensors with little or no storage capacity. The raw communications links proposed for data fusion networks have small error rates, and relatively simple forward error correction methods can be used to reduce the link error rate to insignificant values. Data loss due to network congestion may prove to be a significant factor.

One of the main drawbacks of secret sharing techniques is that they expand the data to be transmitted by a factor of n. This can be broken down into a factor of k to provide security and a factor of n/k to provide reliability. Therefore, after selecting the value of k to provided the desired level of privacy, one is tempted to find the smallest value of n to provide the necessary reliability. A critical issue with this approach is choosing the proper paths over which to send the shares. In the high-speed high demand environment envisioned, global information will generally not be available, hence local or distributed path allocations schemes are needed. A more promising solution appears to be using a large value of n together with a priority scheme that can locally decide to drop a certain number of messages while retaining the required level of reliability. This provides an algorithm that can adapt more efficiently to changes in the network. It is also better matched to the distributed nature of TCP/IP.

Another important aspect to be considered is that of receiver buffering. At the destination, the message can be reconstructed only after the first k messages arrive. Since the messages are sent asynchronously, there may be a large number of other messages arriving before enough shares of the first message are received. Thus the amount of memory needed for buffering is very large. This problem can be overcome by finding secret sharing schemes that allow for preprocessing that reduces the amount of memory needed, or more generally schemes that are well matched to processor memory hierarchies.

For instance, in the very simple (n, n) secret sharing scheme where for a secret message s the first $n-1$ shares, $v_i, \ldots, v_{n-1}$, are random elements of $GF(q)$ and the last share v_n defined so that

$$s = v_1 + v_2 + \ldots + v_{n-1} + v_n \bmod q\,,$$

as soon as a share arrives, it is added to the accumulated sum of the previous shares.

This way, only the sum of the shares received needs to be stored. Unfortunately, it is not clear how to get this kind of nice memory conserving preprocessing for general (k, n) secret sharing schemes.

To alleviate the requirements imposed by secret sharing on the communication system, it may be advisable to combine the secret sharing with encryption in time. The encryption algorithm would be fast, but not very secure. More generally, encryption algorithms can be used in combination with secret sharing schemes that are allowed to "leak" some information, i.e., subsets of less than k shares should not completely determine the secret, but may be allowed to give some information about it. The reduction in security caused by this is compensated for by the encryption in time. This provides more flexibility for the secret sharing schemes in order to enable memory conserving preprocessing with low processing complexity.

For absolutely reliable communications, it is necessary that data be stored at a network node until the data is known to have been transferred successfully to another node, which is usually closer to the ultimate destination. Therefore the network entry nodes to which data sources directly connect must provide storage for sensors without memory.

Two-dimensional (or array) coding techniques can be used to encode information both in space and in time. This is an alternative to secret sharing, where instead of shares of secrets being sent through the different paths, the messages are first encrypted using some new fast encryption algorithms (in very high-speed networks), then error control codes are used through paths. This may provide a better tradeoff between reliability and bandwidth. However, this method seems to require higher complexity codes which may result in higher latency due to processing, unless highly pipelined processing is used.

Once efficient implementations of space-time code diversity techniques are considered for very high-speed networks, the issue of proper network architecture becomes very important, in particular, the network computer memory hierarchy. All optical highly pipelined processing implies that all optical memories are available. More complex routing schemes also require all optical memories. Not coincidentally, two-dimensional array coding techniques are being proposed for parallel optical memories (POMs) [Alan Craig, 1996]. Except for the potential high storage density, currently envisioned POMs are not very well matched to high-speed photonic networks, because the inputs and outputs of POMs are electronic in nature. For instance, a single photonic POM output channel converts 100 photons into an electronic signal by integration within a microsecond—many orders of magnitudes below GHz rates! We have been looking into ways to change this situation, making POMs more useful for all optical networks in general and provide for memory necessary for high-speed coding applications.

4.2 Network Architecture and Processing

We have carried out a survey of network architectures and technologies and identified the current baseline solutions. For example, NTONC (the National Transparent Optical Network Consortium) is proposing a combination of the ATM and SONET technologies. We have identified a number of issues of concern in this solution, e.g., ATM load-oscillations, retransmission of SONET frames on packet errors, network scaling problems, and LAN/WAN/Longhaul mismatch (e.g., 802.X standard vs. OC-48 standard). There are also a number of issues associated with all optical networks: the need for digital regeneration to suppress noise; how to deal with hard network failures due to laser malfunction or drifting out of band; and feedback loops caused by optical oscillations due to erbium amplifiers.

In order to tackle these issues, we have developed an alternative that is intended to alleviate these concerns. Our SUMIT network architecture features space-time code diversity and other coding techniques in order to provide a flexible and scalable photonic network architecture. Security and robustness are supported by a reconfigurable combination of novel photonic quantum devices and fiber technologies. The capabilities of these devices appear to be well matched to support photonic communication (transmission, routing and conversion) and high speed processing (coding, security, data-fusion). However a number of issues still remain to be studied: the wavelengths of the prototype devices must be shifted toward standard communication wavelengths, for instance, using gallium-nitride techniques; the packaging of these devices must be refined, converting the current free-space interconnects to fiber-optic interfaces using microlenses; optical gain can and will be added; and electronic speeds must be improved using microwave design techniques, such as transmission lines and traveling wave techniques.

The optimal choice of network architecture for data fusion networks is influenced by both the data fusion requirements (high speed, multicast, narrowcast, security) and the technology used to implement switching nodes and communications links. Important design choices include network topology, routing methods (circuit switching versus packet switching), multiplexing techniques (e.g., WDM, TDM or CDM), and switching node complexity.

The design of the network topology will be largely determined by both the data fusion requirements and the technology available. To achieve high speed and secure communications, several requirements have to be satisfied. As pointed out earlier, to enable secure communications, several different paths through the network are needed between any two users. Essentially, the minimum number of different paths, or more generally the minimum cross-section bandwidth, between any two users is determined by the level of security needed for the data flowing between them. Two important factors in determining the number of nodes and links required for the network are security and cost. The larger the number of nodes and links, the higher is the security level, but also the total cost. Therefore there is a tradeoff between cost and security.

Traditional methods for determining where to put the nodes and to whom they should be connected will have to be adapted to this new scenario. All-optical networks will not be able to use the same buffering methods as used for electronic networks because efficient random access memories that work at the speeds required for SPINs are not likely to be available in the near future. New strategies for network topology design must be investigated.

The choice of routing techniques has a big impact on the security aspects of data fusion networks, since it determines how the various pieces of information will transit through the network. For instance, the path assignment strategies of circuit switching and packet switching lead to different secuirity considerations.

Simple circuit switching has an apparent advantage for high data rates; once a connection has been established, no processing or storage is needed at the switching nodes. However, there are also a number of disadvantages to circuit switching. First, from a quality of service point of view, additional hardware resources are needed to accommodate the peak traffic and standby equipent is needed for the case of a network failure. Second, there is a necessity for more centralized global control, which is more vulnerable to single point failure. In addition this generates the problem of acquiring and transmitting global state information, which tends to increase latencies. This centralized information would be highly critical and require extremely high level of security, since the system depends on it to make good path allocation decisions, and this global information would allow an outsider to determine the voulnerable points of communication network. Another disadvantage of circuit switching is its unsuitability to extremely variable requests for high rate image data due to its limited resource reallocation capability. One approach to try to overcome this last difficulty is to combine circuit switching with multiplexing techniques.

Packet switching has the advantage of being very flexible in terms of resource allocation. Therefore it is well suited for extremely variable high rate image data. Algorithms for packet switching exist that have minimal need for centralized information. This makes the network extremely robust and capable of adapting quickly to topology changes due to link or node failures. This was one reason for the development of the TCP/IP protocol for the DOD. However, packet switching requires more complex routing techniques to ensure proper message delivery. These routing schemes would have to be combined with the coding schemes discussed in the last section to guarantee the network maintains the desired security levels. Priority schemes are needed to decide resource allocation among the competing message shares. This decision should take into account not only the priority associated with the "importance" of the message itself, but also the importance of the particular piece of information in the message reconstruction process.

For very high data rate optical links, the classical buffering approach to deal with collisions would require either optical storage or optical/electronic conversions; the former does not easily provide sufficient storage, the latter is at present not fast enough.

This problem can be solved by the use of the priority schemes coupled with our "space-time diversity" techniques discussed earlier, where some packet loss can be tolerated at intermediate nodes, since the destination needs only a certain minimum number of them to recover the message. Therefore, the priority scheme has only to guarantee that with high probability the required minimum number of packets associated with a given message arrive at the destination.

The constraints imposed on the priority scheme may be reduced by allowing some memory on the intermediate nodes. One alternative is the use of optical loops to store packets in a ring fashion. Packets are stored sequentially and can only be retrieved sequentially after a round trip delay in the loop, as in circular queues. This suggests an approach that uses the bidirectional network links themselves as temporary storage loops, trading off bandwidth and memory reasources. More generally networks are then composed of loops that are interconnected by routers.

When using a packet switched network, several parameters need to be specified. Of great importance is the choice of packet format and size. One of the main questions related to this is that of header treatment. It would be desirable to encrypt the header to prevent an eavesdropper who happens to obtain a certain packet from knowing where the packet came from and what is its destination. In each node, routing information would be extracted on a "need-to-know" basis. This enhances the security level of the network, since now the wiretapper not only does not know where to look for the packets associated with a particular communication link, but also, he would not know if the packet he intercepts is associated with that particular communication link. On the other hand, this makes it more difficult for the intermediate nodes to find out where the message should be directed, most likely increasing latency.

Another aspect of the packet format to be decided is whether they should have fixed or variable lengths. Fixed length packets are easier to handle from a network point of view, but the variable length are more flexible from a user point of view and may adapt better to the priority scheme suggested above. Finally, the size of the packets is of importance. Big packets have smaller control bit and synchronization overhead and lead to fewer fragmented messages which have less problems with message reassembly. On the other hand, making the packets bigger increases the amount of network resources needed per node to handle them.

There are several multiplexing techniques that can be used in conjuction with both of the switching techniques described above. Each has different characteristics that are more or less suited to one of the switching methods. We are considering the use of several multiplexing techniques—WDM, TDM, CDM, and STDM—in conjunction with these two switching methods.

We are also studying these techniques in the context of communication protocols, such as TCP/IP. Current architectural models can be used to study how current processor architectures can be migrated to deal with the challenges of very high speed photonic networks, such as very high latencies, very high parallelism, and (to begin with) low photonic processing complexities. For instance, our recent studies indicate that routers for very high speed photonic networks can be designed using the very robust and distributed TCP/IP protocol. Most of the processing for TCP/IP involves counters and CRC computations, which appear to map well onto very fine-grain architectures such as FPGAs. Network nodes for our proposed SPINs may incorporate small table lookups, or "photonic FPGAs." The more complex, but less frequent, processing requirements for TCP/IP can be handled with an order of magnitude less processing speed by standard processors. We are also considering VLIW-like architectures that appear to be very well suited to support routing the many channels in very high speed photonic networks. We are studying the various design alternatives, and the necessary tools to support such designs.

From a user perspective, network architechtures should be evaluated using Quality of Service (QOS) measure. Traditionally, the circuit-switched architectures of the Telcos are said to have higher QOS than packet-switched architectures. The QOS of the latter can be adjusted with the use of priority schemes to match the QOS of circuit switched architechtures. It is worthy of note that the deregulation of the Telcos industry in the US has reduced the QOS (at least for some circuit-switched networks) to be comparable to nonpriority scheme based packet-switched networks.

A key aspect of SPINs is network control using very high speed processors. We simulate advanced processor architectures to assess optimum processor parameters and overall network performance. For instance, for a very high speed network using TCP/IP routing on numerous virtual channels, a representative question is what kind of architecture is well suited for such a task?

4.3 Basic Technologies

On the photonic technology side, we have studied a number of device technologies, their physical limits, and their impact on network parameters such as bandwidth and latency. Ideal devices for photonic networks have very high bandwidth, low latency, and good timing properties; are either very linear or level-restoring, are cascadable, convert efficiently between photon and electron energies, have low noise, are small, and are easy to integrate; and last but not least are inexpensive. Clearly none of the current devices support most of these properties. Some of the devices lead in some of the categories, a few lead in none. The architectural challenge is to select, mix, and match the right photonic technologies in order to meet the requirements of very high speed secure networks.

These devices can be characterized by the bandwidth and latencies of the optical data path and the bandwidth and latencies of the control, either electronic (for electro-optical) or optical (for "true all optical") input. Many devices suffer from long latencies, recovery periods, or slow integration times that have to be "hidden" by suitable network architectures.

Examples of *control latencies* are: Acousto-Optic Modulators (~1μs), TOAD (~10 ns), Sagnac loop (at 100Gb/s, ~1 ns), electronically controlled MQW (Multi Quantum Well) devices (~20 ps), inter-subband transition based devices (~1ps), virtual-transition based devices (e.g., χ_3, <1 ps). As NTT has demonstrated, the Sagnac loop can be used to make the fastest single links to date at 200Gb/s, while still supporting WDM in addition. For an overview comparing bandwidth and latencies of various networks and device, see Figure 1. We plan to expand this type of analysis to include more devices, such as lithium-niobate couplers.

We are attempting to go one step further by reporting back to the devices development effort what would be desirable devices from an architectural point of view, and participate in the process of developing and evolving such desirable devices. One of the advantages of this process is that global optimization and tradeoffs become possible in shaping devices, subsystem, and system designs. As an example of how system and device design can interact, it has been proposed to use tuning of the lasers and the dispersion properties of optical fibers to align packets from different sources.

In order to test such global optimization and tradeoff strategies we are cooperating with quantum well device developments at Stanford to build 3-D photonic quantum devices, including stackable MQW mirrors, quantum wires, and quantum dots. Figure 2 illustrates a vertical cross section of a 3-D stack of quantum dots. GaAs quantum dots have recently been grown directly on silicon, opening up a revolutionary opportunity to transform silicon CMOS into a photonic technology. We are still studying how these capabilities can be used.

The main breakthrough of this approach lies in how the structure of these photonic devices and architectures take advantage of the parallelism afforded by space, time, frequency, and code division multiplexing. For instance, by stacking these quantum devices in arrays reminiscent of systolic arrays of early VLSI architecture proposals [Computer, 1982]. Our approach to global optimization of architectures includes all levels down to individual photonic switches rather than classical logic functions, resulting in much more robust system designs, as reported in [Powell et al., 1996]. Individual systems or subsystems can then be fabricated to suit particular communications needs. For example, a flexible, table-directed architecture can be developed at the hardware level by organizing the structure of the photonic quantum devices into reconfigurable gate arrays that are logicaly lookup tables and physically implemented as trees of multiplexers. Crossbar switches can then consist of simple combinations of energy/information conservative switching elements, such as X-type gates.

The devices we studied can also be used for routing and interconnection of systems level components. The functionalities of these devices are designed to match photonic communication system needs (transmission, routing, switching, data conversion) and high speed data processing (coding, security, image transmission, and data fusion). We created two new fundamental switches: the X-modulator (shown in Figure 3) and the vertical-cavity phase-flip modulator (VP-modulator); see [Trezza et al., 1996]. The X-

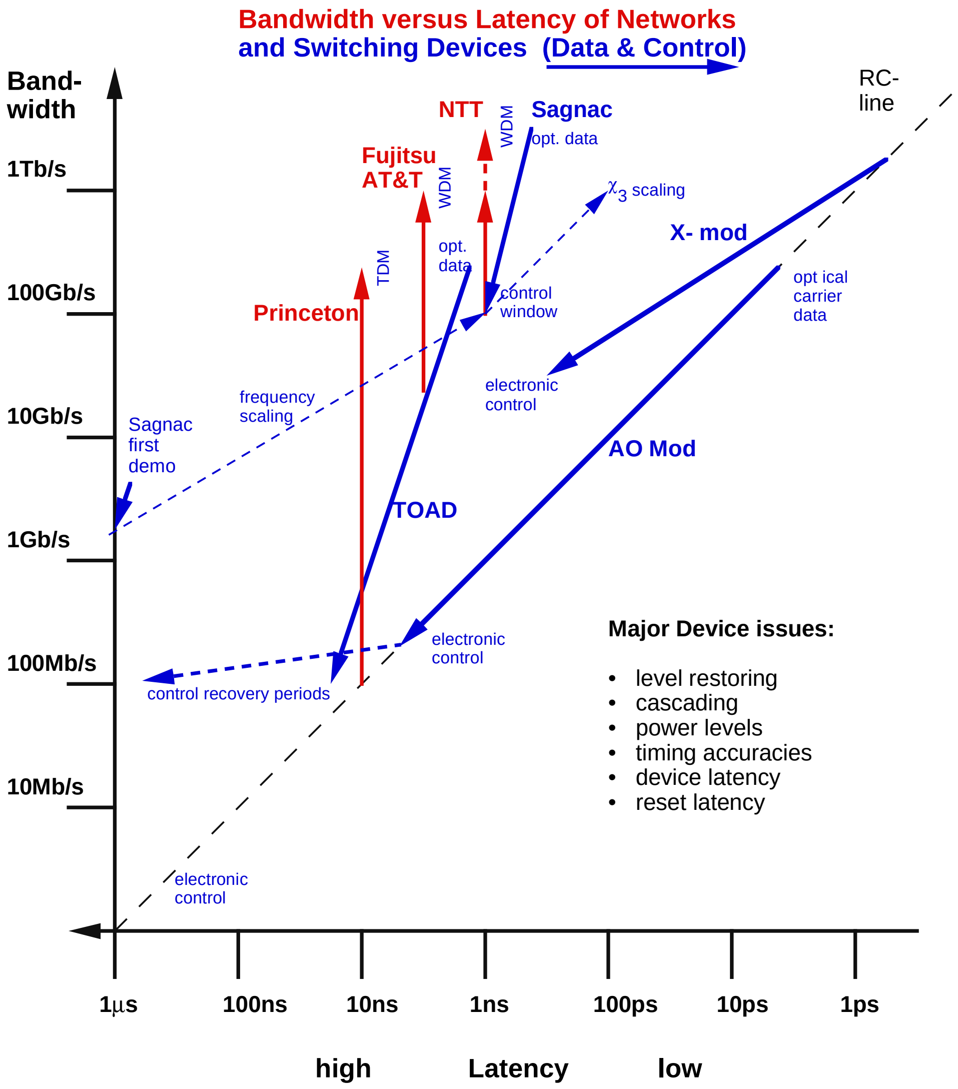


Figure 1: Bandwidth versus Latency of Networks and Switching Devices (Data and Control)

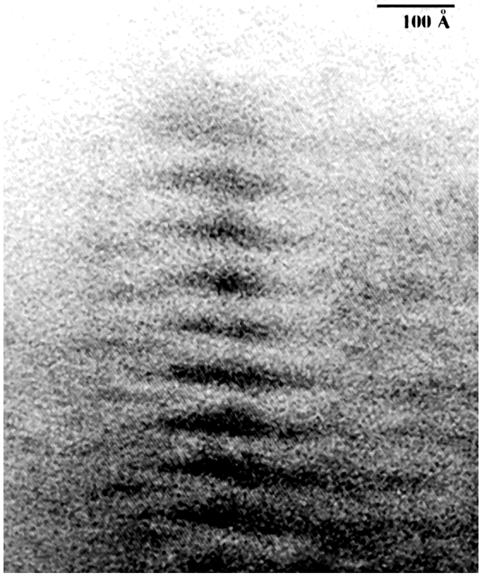
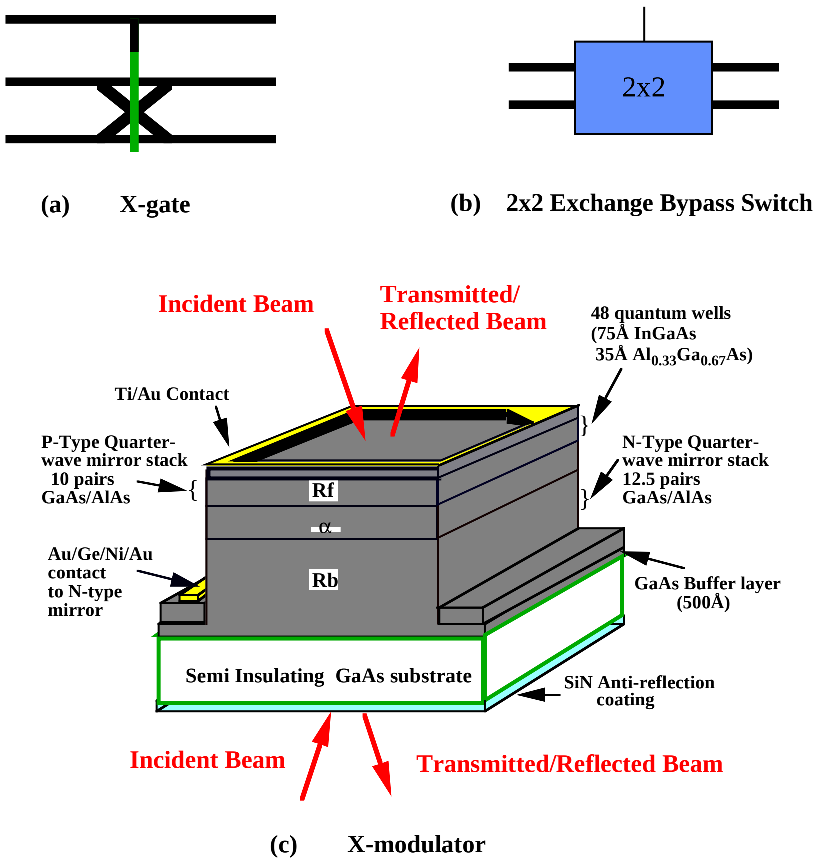


Figure 2: Vertical Cross Section of 3-D Stack of GaAs Quantum Dots

modulator has a primitive functionality that directly implements the exclusive-or and exchange function, while the VP-device controls the phase in order to steer optical beams.

An important physical level feature of both devices is that photons are neither destroyed or created, which saves energy and reduces cooling considerations. At the higher level, thse devices support higher order (information preserving) functions that improve the efficiency and performance of communications tasks since large optical outputs are obtainable in both switching states (on/off) of the devices. This latter attribute allows cascading, stacking and interconnection; hence complex 2-D and 3-D architectures can be built with these quantum devices. Furthermore, arrays of devices can be created to solve computation problems, and system level problems involving photonic interconnections. The unique properties of these devices also show up in the way physical properties such as device losses are connected with higher level properties such as logic conservation. The simplicity, small size, low power requirements, and high fabrication yield of the individual devices allows the design and fabrication of large complex systems, while optimizing systems yield and therefore minimizing systems cost.

We have considered a number of different system configurations that are experimentally accessible to study as a direct result of this work on quantum devices. We have developed approaches to realize very high speed devices, from all optical (analog) modulators to all optical (data and control) digital gates. We studied the impact of these devices on functionality and performance of parallel systems designs.



(a) X-gate

(b) 2x2 Exchange Bypass Switch

(c) X-modulator

Figure 3: X-Modulator Device

Our design capabilities allow us to investigate quantum wells utilizing the entire range of the {In, Ga, Al, As} materials system. While investigating multiple quantum wells, we have designed and experimentally verified exciton systems that provide absorptive modulation at a variety of wavelengths ranging from 632 nm (HeNe wavelength) to 1300 nm and 1550 nm, important to fiber optic communications wavelengths.

4.4 Appendix: VLIW Approaches to Network Processors

High-performance processor design has recently taken two different (and often diametrically opposed) approaches. One approach used to design high-performance processors is to increase the execution rate by increasing the clock rate of the processor or by reducing the latency of operations. Another approach is to issue and execute multiple operations concurrently. Traditional processor designs that issue and execute at most one operation per cycle have been referred to as "scalar" designs. Processor designs that can issue and execute more than one operation per cycle will be referred to as *super-scalar* processors.

Historically, two primary techniques have been used to achieve super-scalar performance. The first technique uses dynamic analysis of the instruction stream during execution to determine those operations that are independent (processors that use this technique are commonly referred to as superscalar processors). These independent operations can be issued and executed concurrently. The second technique uses static analysis performed by the compiler to schedule independent operations together into compound multi-operation instructions (processors that use this technique are commonly referred to as VLIW processors). All operations in a given instruction can be issued and executed concurrently with no dynamic analysis.

Neither superscalar nor VLIW architectures are entirely satisfactory. Superscalar processors require complex control logic to determine dynamically the operations that are issuable, which limits the amount of parallelism that can be exploited as well as the performance of the processor. VLIW processors are not adaptable to unpredictable or variable system behavior, which is especially a problem when dealing with non-uniform memory architectures. Considering the characteristics of both superscalar and VLIW processors, it is clear that an ideal processor would have the advantages of both and the liabilities of neither.

This research has developed an architectural model as well as a simulation tool that is being used to evaluate performance variations across a design space that spans both superscalar and VLIW architectures. By varying parameters to both the compiler and the simulator, many different processor configurations can be simulated and their performance compared. The basis for this model is the *Split-Issue* execution paradigm which separates the architectural (or *virtual*) behavior of an given machine (behavior that is exposed to the user) and the implementation (or *real*) behavior (behavior that is native to the particular implementation but is not exposed to the user).

4.4.1 An Introduction to Split-Issue

The execution of an operation typically consists of three distinct phases: the acquisition of source values, the computation of result values, and the delivery of the completed results. In a traditional pipelined processor, these are often described as *register fetch*, *execute*, and *register write-back*, and are considered to be inseparable steps in the execution of an operation in the pipeline. In the Split-Issue model, these phases are decoupled from each other so that they can be treated independently in the implementation. In order to maintain the correctness of the virtual machine, each event is scheduled to take effect at the appropriate time based on the architectural specification for each operation.

Whereas in the traditional pipeline, source, result, and intermediate values are held in latches in pipeline stages until conditions are set to proceed, in the Split-Issue model these are held in temporary storage locations.

The original concept for Split-Issue was presented in [Rau, Micro-25, 1992] and described more completely in [Rau, Micro-26, 1993] as a mechanism for supporting a dynamic execution model for a VLIW processor capabilities. In the Split-Issue model, described in detail in [Rudd, SU-TR-657, 1994] and [Rudd, Asilomar, 1995], operations are split into into three independent phases: *PhaseOne* performs all acquisition events, *ExecutePhase* performs all computation, and *PhaseTwo* performs all delivery events. The PhaseOne and PhaseTwo events are similar in that they are access events performing read and write accesses to the register file and are synchronized to maintain the virtual ordering of operations as scheduled by the compiler. The ExecutePhase events are computations that operate on the operands that have been fetched by the PhaseOne events and produce results that will be written back by the PhaseTwo events. Unlike the PhaseOne and PhaseTwo events, the ExecutePhase events operate asynchronously whenever their source values are available—all synchronization is maintained by the other two phases. A given operation can be thought of as a triplet of events that will be processed by each of the three queues. For example, in this model, an operation defined as $C \leftarrow f(A,B)$ results in the triple event:

$$\begin{array}{lll} PhaseOne: & T_1 \leftarrow A@0 & T_2 \leftarrow C@0 \\ ExecutePhase: & T_3 \leftarrow f(T_1,T_2) & \\ PhaseTwo: & C \leftarrow T_3@2 & \end{array}$$

Note that the ExecutePhase events do not need any timing information specified, since they are able to commence once all operands are flagged as available.

Arbitrarily complex operations can be constructed in this manner and even the most complex CISC processor operation can be framed in this form. For example, even indefinite iteration can be specified as in $T_x, \ldots \leftarrow h(T_x, \ldots)$. In this case, the temporary value T_x is used to force recomputation until it is assigned an appropriate value to mark the computation as completed. One tangential use of Split-Issue techniques might be the emulation of CISC processors. Emulating a CISC processor by mapping the CISC operations (the architectural operations) into native operations has recently come into vogue in two commercial microprocessors—both the AMD K5 and the Intel Pentium Pro exploit the use of distinct virtual and implementation architectures. In fact, simply by changing the mapping between the virtual and implementation architectures, it is possible to have a single implementation that can emulate different virtual processors.

An observation of the representation used in the above two examples is that it bears a striking resemblance to the information contained in a machine specification for the operations. In a machine readable form, this is essential information that a compiler would need to schedule the operation and is also essential information that a processor designer would require to implement the same operation. This natural connection

Table 1: Simulation Parameter Summary

Parameter	Description
Pipeline stage limits	Each pipeline stage has different limits that can be used to represent the characteristics in performance or local storage.
Scheduling-unit makeup	Individual scheduling structures may be combined to better schedule operations to available function units and their performance may be varied to represent a range of scheduling capabilities. Additionally, the basic characteristics of each phase can be adjusted independently.
Function-unit makeup	The number and capabilities of function-units of each type may be varied to provide a range of computation capabilities.
Load-store ordering	Different memory ordering restrictions may be used to restrict the dynamic reordering of loads and stores.
Memory system features	The size and characteristics of separate instruction and data caches, performance of the bus, and latency of the memory system may be varied.

between specification and model is fortuitous—and the next section demonstrates that there is a similar connection between the model and implementation as well.

4.4.2 The NV Simulator and Experiment Environment

The simulation environment used in this research is the NV (Nyfo-VLIW) simulator, which is part of the University of Illinois (Urbana–Champaign) IMPACT compiler suite. It is based on the Split-Issue model and is parametrized to allow the simulation of a wide range of system configurations and models both the processor and memory systems. The table of simulator parameters lists a number of the areas that are parametrized in the simulator. In addition to the capabilities listed in the table, the compiler is parametrized and supports any configuration of virtual function units and schedules code based on the characteristics of the available function units.

The processor architecture being modeled is that of a fixed-width VLIW processor—this is the architectural processor. The actual configurations that are used in the experiment vary from a traditional static VLIW processor to a complex out-of-order processor that can have many operations outstanding and dynamic scheduling of available operations.

We are in the process of running a number of SPEC92 and Unix benchmarks under the simulator, varying a number of the parameters to understand better the performance implications changing different aspects of the model. The current experiments consist of varying the number of function units, the number of scheduling structures, and the main memory latency to see how the performance varies over these changes. The initial results have shown that, as expected, as the number of function units per scheduling structure increases the utilization of the function units improves. They have also shown that, again as expected, the number of outstanding operations allowed also increases the utilization of the function units. Both of these situations result in the increase of the operation pool available to a given function unit, and thus improve the possibility that an operation will be available to issue to that function unit.

Future work in this area includes addressing both compiler as well as architectural issues. The current architectural model supports unconditional in-line code execution only. This is a significant limitation on performance and future work will include the ability to have the compiler schedule operations that are predicated and non-excepting for static speculation as well as having the processor follow predicted branch paths for dynamic speculation.

4.5 Appendix: Fundamental Limits on Optical Switching and Memory

Photonic devices are generally larger than electronic devices since they are larger than the wavelength of light ($\lambda \approx 0.5\mu$). While quantum well (QW) devices can provide light signal sources with a spacing well under λ, detecting such a signal seems to require devices larger than λ. This open issue has important systems implications. Any detector array that resolves light patterns and can be packed into a space less than λ is referred to as a *super-resolution* device. Our research focused on the possibilities of the realization of such super-resolution devices. Fundamental limits may or may not lead to practical limitations for a particular application or a class of applications. As the term "photonic" implies, the primary focus here is on photons, but charged carriers in solids are also involved. In particular, there are photon-electron interactions that involve various transitions, from relatively slow interband-, bandgap-, and intersubband-, to the fastest virtual transitions (e.g., χ_3).

Quantum mechanical limits The Heisenberg uncertainty principle limits the spatial extent of single photons. With respect to the transverse position x, this principle states that the product of the uncertainty in the position Δx and the uncertainty in the momentum Δp_x cannot be less than $h/4\pi$, where h is Planck's constant. Because the momentum of a photon is directly proportional to its wave vector, $k = 2\pi/\lambda$, this leads to the statement that a single photon is diffraction limited.

Diffraction limit and super-resolution The spatial resolution of any optical system is determined by a number of factors relating to specific design parameters. The fundamental limit that is common to all unguided systems (free-space propagation) is the diffraction limit, which specifies the limiting spatial angular resolution θ in terms of the transverse extent (aperture size) D of the system and the wavelength λ of the light. The limit θ given by $\sin\theta \approx \lambda/D$, where in order to achieve equality we must include a coefficient of order unity that is dependent on geometric details. However, the diffraction limit is not truly an absolute one on a macroscopic (multi-photon) scale, and systems can in principle be constructed that violate this limit. The reasoning behind this, based on the mathematical theory of analytic functions, is discussed by [Goodman, 1988]. As a practical matter, however, the diffraction limit is generally accepted as a standard, and resolution beyond this limit is difficult to achieve, primarily due to the fact that any detection system that is used to determine an analytic representation of the image to be resolved introduces noise. Therefore, the design of a super-resolution system, that is, one with resolution beyond the diffraction limit, involves the need for low-noise methods. These methods typically utilize temporal or spatial integration or both, so some compromise with speed requirements may be anticipated. In recent years, the application of a variety of super-resolution methods to radio direction finding [Gething, 1991], to optical imaging [Hunt, 1995], and to other fields has attracted increasing interest.

Guided wave limits When light propagates in a waveguide, the electromagnetic boundary conditions at the surfaces of the waveguide determine the nature of the propagation [Ramo et al., 1965]. In particular, the conducting rectangular waveguide, commonplace for microwaves, limits the larger of the two transverse dimensions to be at least one half wavelength in width. If the width is less than this, the light will not propagate, but rather will decay exponentially along the length of the guide, so that after a very short distance no power remains. It should be noted that because this limit is one-dimensional, the other dimension may be arbitrarily small. This flattened geometry is particularly relevant to solid-state devices.

Thermodynamic limits Thermodynamics places a limit on the size of a device only indirectly through the noise level. As the size of the system decreases, the amount of energy associated with one bit of information must decrease also. For a fixed wavelength of light, the number of photons must decrease, with an absolute minimum of one photon per bit. Similar statements can be made for electrons.

Classical Rayleigh scattering Quantum dots of radius a are much smaller than the wavelength λ of the light. In the usual formulation of Rayleigh scattering, it is only the far field region that is of interest, the region that is very far away from the scatterers. However, in the present case, because of the small propagation distances anticipated, we might also be interested in the near field region. The distinction between these two regions is generally stated in terms of the dimensionless Fresnel number [Saleh and Teich, 1991], $N_F = a^2/\lambda z$, where z is the propagation distance from the scatterer to an observer or detector. The near field extends over small distances up to the point where the Fresnel number is of order unity. If we set $N_F = 1$ in order to determine the value of z that separates the near field from the far field, then this equation shows that the ratio z/a of propagation distance to dot radius equals the ratio a/λ. Since this ratio is much less than unity, propagation over distances comparable to a dot radius will be clearly in the far field. This conclusion must be regarded as tentative, however, as the conditions under which the very common Fresnel diffraction and Fresnel numbers can be used may not apply when the quantum dots are much smaller than the wavelength.

Encoding with quantum dots One approach to packing a large amount of information into a space that is small in comparison to the optical wavelength of light beam is based on the idea that high spatial resolution is not required for high packing density of bits. Resolution of detector sensitivity can suffice for this. To see this, consider the following example.

Suppose we pack a 16×16 planar array of quantum dots into an area that is much smaller than $\lambda \times \lambda$. In this case, neighboring quantum dots cannot be resolved optically because of the diffraction limit. Nevertheless, as the quantum dots can be made to absorb (or scatter) light, if a detector is placed in the light beam behind the plane of the dots so that it collects the transmitted light, the strength of the detector signal will be reduced in proportion to the number of dots that are *switched to the absorbing state*. Better yet, if the dots can be made to reflect the light, then the detector can be placed to the side but in front of the plane of the dots, so that it receives only reflected light. Then the detector signal will be proportional to the number of dots that are *switched to the reflecting state*. In this example, there are 256 levels, so this array can transmit eight bits in parallel. Clearly this approach is limited in its efficiency, since the number of bits is only the base-2 logarithm of the number of lines, but it does provide a simple method for increasing the packing density beyond the diffraction limit.

5 Technology Transfer

We keep a number of outside organizations and individunals up to date on our results by personal communication, e-mail, conference presentations and journal publications.

John Trezza, who created the first X-modulator, successfully transferred this technology to industry after graduating. He is now working at Saunders-Lockheed–Martin in Boston.

Bibliography

[Blakly, 1979] G.R. Blakly. "Safeguarding Cryptographic Keys," Proceedings of the National Computer Conference, AFIPS, Vol. 48, 1979, pp .242–268.

[Computer, 1982] *Computer*, Special Issue on Highly Parallel Computing, January 1982.

[Alan Craig, 1996] Workshop on Data Encoding for Page-Oriented Optical Memories (DEPOM'96), March 27–28, 1996, Phoenix, Arizona. Sponsor: Alan Craig, AFOSR.

[Gething, 1991] P.J.D. Gething. Radio Direction Finding and Super-resolution. Peregrinus, 1991.

[Goodman, 1988] Joseph W. Goodman. Introduction to Fourier Optics. McGraw-Hill, 1988.

[Hunt, 1995] B. R. Hunt. "Super-resolution of Images: Algorithms, Principles, Performance," International Journal of Imaging Systems and Technology, Vol. 10, 1995, pp. 297–304.

[Karnin, Greene, Hellman, 1983] Ehud D. Karnin, Jonathan W. Greene, Martin E. Hellman. "On Secret Sharing Systems," *IEEE Transactions on Information Theory*, Vol. IT-29, 1983, pp. 35–41.

[Powell et al., 1995] J. S. Powell, J. A. Trezza, M. Morf, and J.S. Harris, Jr. "Vertical Cavity X-Modulators for WDM," In *MPPOI'95*, October 1995.

[Ramo et al., 1965] Simon Ramo, John R. Whinnery, and Theodore van Duzer. Fields and Waves in Communication Electronics. Wiley, 1965.

[Rau, Micro-25, 1992] B. Ramakrishna Rau. "VLIW: Not Your Father's Oldsmobile," Invited talk, Micro-25, Dec. 1992.

[Rau, Micro-26, 1993] B. Ramakrishna Rau. "Dynamically Scheduled VLIW Processors," Micro-26, Dec. 1993, pp. 80–92.

[Rudd, SU-TR-657, 1994] K. W. Rudd. "Instruction Level Parallel Processors—A New Architectural Model for Simulation and Analysis," Stanford University Technical Report CSL-TR-94-657, Dec. 1994.

[Rudd, Asilomar, 1995] K. W. Rudd. "A Hybrid Architectural Model to Exploit Instruction Level Parallelism," Twenty-First Annual Asilomar Microcomputer Workshop, April 1995.

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

[Trezza et al., 1996] J.A. Trezza, M. Morf, and J.S. Harris, Jr. Creation and Optimization of Vertical Cavity X-Modulators. *IEEE Journal of Quantum Electronics* 32(1):53–60, January 1996.