Issues in Optoelectronic Computing
System Architectures

ITC-98, June 24, 1998

Donald M. Chiarulli
University of Pittsburgh
Pittsburgh, PA, USA

Collaboration: Steven Levitan

Support: United States Air Force Research Labs
Question

How can Optical and Optoelectronic Technology be used most effectively in a computer system architecture?

Three Examples
• Optoelectronic Cache Memory Interface
• Superscaler Processor w/Chip Level Optical Interconnections
• Reconfigurable Processor Architecture
Optoelectronic Cache Memory Interface

Use spatial parallelism at the memory interface to improve overall performance of the memory hierarchy by lowering the effective access time.
Average Memory Latency Comparison

![Graph showing average memory latency comparison for different operations across various memory sizes. The graph plots latency in nanoseconds on a logarithmic scale against memory size in kilobytes. Different operations include Matrix Multiplication (electronic and optical), Heap Sort (electronic and optical), and Image Convolution (electronic and optical). The graph illustrates the performance differences across electronic and optical memory systems.]
Optoelectronic Cache Memory Prototype

System to place a Page-Oriented Optical Memory into the memory hierarchy of a personal computer for high speed data-base applications.
System Block Diagram

- Off-the-shelf Personal Computer (Gateway 2000 P5-166)
- Optical memory implemented as PCI bus add-on memory controller

- Pentium Processor
  - Cache controller
  - Primary cache
  - Secondary cache

- PCI bus bridge
  - Primary memory

- OE cache
  - Detector array
  - Optical memory
    - 128x128

- EISA bus bridge
  - Swapping disk
Spectral Reed Solomon Encoding and Decoding

ENCODER

FFFT\(^{-1}\)

DECODER

FFFT

ERR POLY

ERR VECT

Berlekamp Massey Algorithm

Recursive Extension

optical memory
Flow diagram

Raw data → FFT⁻¹ → \( v_1, v_2, v_3, \ldots, v_{15} \) → FFT → \( D_1 + E_1, D_2 + E_2, \ldots, D_9 + E_9 \) → Find error locator polynomial → Calculate error vector

\( E_{10}, E_{11}, E_{12}, E_{13}, E_{14}, E_{15} \) → Error vector

\( L_1, L_2, L_3 \) → Recovered data + errors → \( D_1, D_2, D_3, \ldots, D_9 \) → Corrected data
Error Correcting Code Performance

Original → Encoded → 1% noise added → Decoded/uncorrected → Corrected
Performance Data (Placed and Routed Design Simulation)
Optoelectronic SuperScaler Processor

Use Chip-to-Chip optical interconnect with fanout to increase the available resources for parallel execution.
von Neumann Architecture w/ Superscaler Processing
3-Chip Optoelectronic Superscaler Processor using CMOS/SEED Technology
Single Chip SuperScaler Processing Element with Chip Level Optical I/O
Optical Setup for OE SuperScaler processor

Reflective Interconnect 1:4 CGH
Spot Array Generator
Laser

Polarizing Beam Splitter
\( \lambda/4 \) plate

ALU Chips Mounted on Common Carrier

ALU Chips Mounted on Common Carrier
Test Silicon
Performance Data
Optoelectronic Dynamically Reconfigurable Processor

Use spatial parallelism in Chip-to-Chip optical interconnect to deliver large volumes of configuration data to the core logic of a reconfigurable processing chip.
Commercial Reconfigurable Logic Devices - FPGAs

i.e. Xilinx 4000 series

Configuration Memory

Serial Configuration: once at power-up

Configuration bits specify logic function and interconnect

Configurable Logic Block

Configuration Buffer
Dynamically Reconfigurable Logic Devices

Program segment ---> Logic ----------> Configuration sequence

a = b \times c

\quad d = e + f

\quad g = a - d

Problem: Deliver large amounts of configuration data at high bandwidth
Optoelectronic Reconfigurable Processor Architecture

Configuration Memory

Array of VCSEL Transmitters

Reconfigurable Processor

Array of Photo Detectors
Fundamental Paradigm Shift in Processor Architecture

von Neumann Architecture with Superscalar Processing

Instruction Stream

- Load (32-bits)
- Sub (32-bits)
- Add (32-bits)
- Mult (32-bits)

Instruction fetch unit

Fixed function execution pipelines

- Integer
- Integer
- Floating Pt.
- Floating Pt.

Dynamically reconfigurable processor

Configuration Blocks

- $10^4$-$10^6$ bits
- $10^4$-$10^6$ bits
- $10^4$-$10^6$ bits

Configuration generator

Reconfigurable processing array
Summary and Conclusions

♦ Optics has a role in systems architecture

♦ Spatial and Spectral bandwidth is as important as temporal bandwidth

♦ Fanout is important

♦ Optics must be applied to specific problems and application domains.