Breaking the Memory Bottleneck with an Optical Data Path[1]

Jason E. Fritts and Roger D. Chamberlain
jefritts@cs.wustl.edu, roger@ccrc.wustl.edu
Computer and Communications Research Center
Washington University, St. Louis, Missouri

Abstract

This paper demonstrates the capability of optical buses in enabling orders of magnitude greater bandwidth between the processor and off-chip memory in a uniprocessor computer system. Through a simulation-based performance analysis of a 1 GHz processor model, we provide a preliminary evaluation of the benefits of an optical processor-to-memory bus in both eliminating the bandwidth bottleneck and in reducing the impact of the increasing processor-to-memory latency gap. The optical technology is constructed of two-dimensional arrays of lasers and detectors bonded to silicon that provide high-speed optical I/O on and off chip. These chip-to-chip light paths may be designed using either rigid free-space optics or flexible fiber image guides. Utilizing the optical data path between the processor and memory provides significantly greater bandwidth with no appreciable latency penalty. We assess the performance impact of this architecture enhancement on a number of media applications. Overall we found that the increased bandwidth nearly eliminates the transfer time between processor and memory, effectively reducing degradation from off-chip memory latency by 50% on average. Additionally, substantial extra bandwidth remains for more bandwidth-intensive architectural options like aggressive latency hiding techniques and single-chip multiprocessors.

Keywords: optical bus, bandwidth bottleneck, performance evaluation, processor-memory gap, media processing

1. Introduction

With the ever-increasing speed of processors, the gap between processor performance and memory performance continues to widen. This performance gap includes two principal factors, latency and bandwidth. The growing processor-memory latency gap has been the primary factor of concern to researchers, and has resulted in the proposal of numerous techniques for mitigating the impact of longer latencies, including lockup-free caches, data speculation, cache-conscious load scheduling, hardware and software prefetching, data reorganization, multithreading, value prediction, and instruction reuse. Many of these techniques can substantially reduce latency penalties, but most of these techniques have the consequence of increasing the process-memory bandwidth, which is also beginning to become a critical problem.

Burger et al. [1] studied the impact of memory latency and bandwidth on overall performance and concluded that if aggressive latency tolerance techniques are implemented, limited off-chip bandwidth may seriously degrade system performance. They also compare processor performance vs. off-chip bandwidth over 2 decades, showing the rate of growth in processor performance far exceeding that of off-chip bandwidth. At the current rates, bandwidth will shortly become a critical bottleneck in many applications. Consequently, future systems will likely see substantial gains if off-chip bandwidth can be dramatically increased.

In this paper, we describe, and evaluate via simulation, a system design that exploits optical technology to improve the bandwidth of the processor-to-memory data path by several orders of magnitude. Two-dimensional arrays of lasers and detectors are integrated with silicon technology, currently providing bandwidths of 256 Gb/s, and likely enabling bandwidths of 1 Tb/s or greater. By constructing an optical light path between the processor chip and the memory controller chip, the bandwidth bottleneck to memory can be effectively eliminated. Furthermore, the increased bandwidth can be coupled with aggressive latency-hiding techniques that diminish the impact of latency on overall performance. The result is a system design that succeeds in breaking the memory bottleneck in current and future computer systems.

This paper provides a preliminary investigation of the advantages of using optical buses for the processor-to-memory data path. Following a discussion of the optical technology in Section 2, the

[1] This research is supported in part by DARPA under grant DAAL01-98-C-0074.

paper proceeds with a series of simulation experiments evaluating the current and future benefits of optical buses. The experiments first examine the benefit of optical buses in reducing long external memory latencies in current processors, and then estimate the performance advantage of optical buses in reducing the growing processor-to-memory gap in future computer systems. The paper's experimental portion starts in Sections 3 and 4 with a description of the base processor model and evaluation environment. Section 5 then describes the experiments and discusses the results of these experiments with respect to our benchmark of media applications. Section 6 introduces some additional motivations and architectural options that may significantly benefit from optical buses. And finally, we summarize our conclusions in Section 7.

2. Optical Technology

The dramatic bandwidth advantages of optics have been exploited extensively in communications networks, where the distances are large, involving a few meters to several kilometers. However, optical technology has not yet been effectively used in boardb-level systems, where the distances are measured in fractions of a meter. Recent advances in electro-optical technology enables the bandwidth advantages of optics at this smaller scale.

A primary enabling technology for this system is the availability of 2-dimensional arrays of Vertical Cavity Surface Emitting Lasers (VCSELs) and photodetectors bonded to silicon circuitry [2]. The technique is illustrated in Figure 1, which shows a 2×2 array of VCSELs and a 2×2 array of detectors bonded to the surface of a CMOS chip. Unlike older edge-emitting lasers, the VCSELs transmit their light vertically, out of the plane of the chip. The data rate achievable in each light path is significant (1 Gb/s or better), and as the number of lasers and detectors grows, the result is an optoelectronic data pathway that provides orders of magnitude greater off-chip bandwidth than traditional electrical pins.

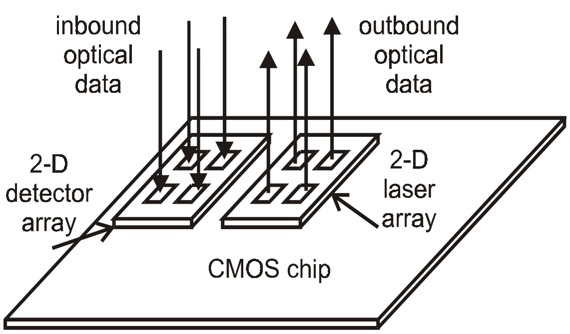

Figure 1. Optical I/O at the chip level.

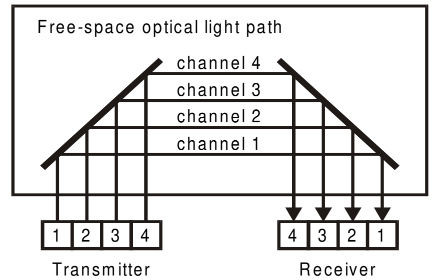

Figure 2(a). Rigid free-space optical link.

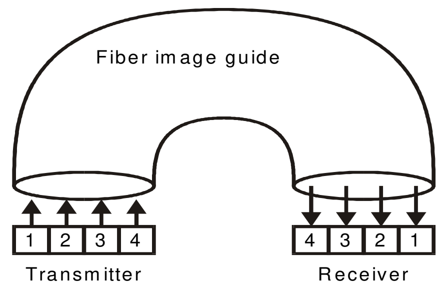

Figure 2(b). Fiber image guide optical link.

With current thresholds below 1 mA for modern VCSELs, the laser power is manageable, and the Metal-Semiconductor-Metal (MSM) technology necessary for the photodetectors is fairly mature. The union of silicon processing with GaAs-based optoelectronics provides a powerful combination, significantly increasing the communications bandwidth available off-chip.

Prototype chip-to-chip optical interconnects have been constructed by Plant et al. [3] with 16×16 arrays of VCSELs and photodetectors, and 32×32 arrays will soon be available. In these systems, the VCSEL and photodiode arrays are flip-chip bonded to CMOS chips using heterogeneous integration techniques. Although Plant's demonstration used bulk optics to deliver light between ICs, the free space optical path for a viable system design could use either a rigid optical link [4] (useful for chip-to-chip links on a board), shown in Figure 2(a), or a flexible fiber imaging guide [5,6] (useful for board-to-board or chip-to-chip links), shown in Figure 2(b).

On chip, the size associated with high-speed (> 1 Gb/s) laser drivers and receivers is small enough to fit in a 125 μm × 125 μm area [3], smaller than half the area typically required for a traditional electrical pad and associated drive circuitry. Combine this with the fact that the I/O is no longer restricted to the periphery of the chip, and the data rate available on and off chip is dramatically increased over all-electrical techniques. With a 32×32 array operating at 1 Gb/s per laser, the aggregate data rate is 1 Tb/s.

Figure 2(a) shows a side-view conceptual diagram of a rigid free-space optical link providing a one-way link from a transmitter chip on the left to a receiver chip on the right. Four VCSELs in the transmitter array are illustrated in the lower left of the figure. The other dimension of the 2-D VCSEL and detector arrays is orthogonal to the page. The light travels vertically off the transmitter chip, is redirected by two 45° mirrors, and is imaged onto the detector array on the receiver chip. Since the optics are inherently bi-directional, the link can be made bi-directional by placing both VCSELs and detectors in the 2-D array on each chip.

The demonstration of a link for a 16 × 32 array (with interdigitated VCSELs and detectors) is described by Chateaunneuf et al. [4]. In this system, 2-D microlense arrays are used above the individual VCSELs to collimate the beams, and mini-lense arrays are used horizontally between the mirrors to provide tolerance to misalignment (an important consideration for commercial viability). The design is compatible with manufacturing via molded plastic optics, an important cost consideration.

Note that for the rigid optical link, shown in Figure 2(a), the two optically-connected chips reside in the same plane. This is commonly the case when performing chip-to-chip communications on the same board. An alternative option is to provide an optical path that is more versatile, via a fiber image guide, as illustrated in Figure 2(b). The flexibility of the fiber image guide enables the endpoints of the link to be in arbitrary orientation relative to one another, making this option appropriate for board-to-board communications.

Fiber image guides are constructed with a large number of individual fibers packed closely together with thin cladding layers. Typical dimensions are 10 µm core diameter and 12 µm outer diameter for the cladding [6]. With an image spot from an individual laser on the order of 30 µm, the light is coupled into a number of neighboring fibers. At the receiver, the beam shape impinging on the detector is approximately a discretized Gaussian distribution, providing excellent coupling into the detector. Practical systems have been constructed by simply butt-coupling the fiber image guides to the laser and detector arrays [6], keeping the implementation complexity (and therefore cost) down relative to systems that require complex coupling optics.

We have previously described and modeled the performance of embedded multicomputer systems that exploit the above optical technologies in the inter-processor communications network [7,8,9,10]. Here, we are interested in taking advantage of the bandwidth provided by optics in the processor-to-memory data path of an individual processor.

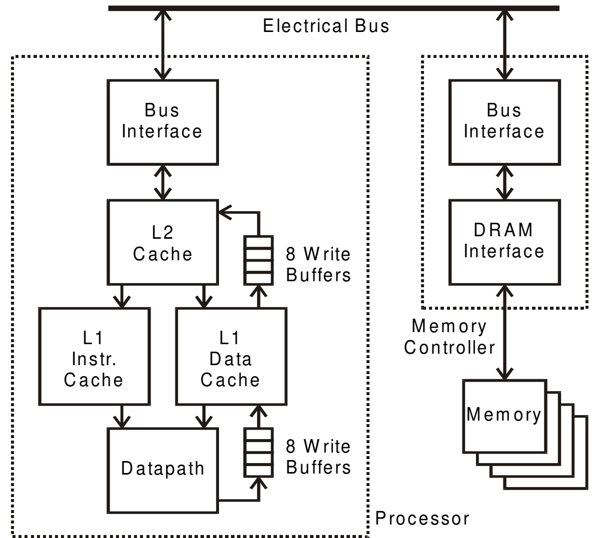

Figure 3. Base computer model with electrical path to memory.

3. System Architecture

Systematic performance evaluation requires a base processor model for comparison purposes. The base processor defined here (shown in Figure 3) is a 4-issue processor targeting 1 GHz operation, with instruction latencies scaled up from the Alpha 21264 (see Table 1) [11].

The memory hierarchy of this processor model provides separate L1 instruction and data caches, a 256 KB unified on-chip L2 cache, and an electrical bus to memory that operates at $^1/_8$ the processor frequency and supports split bus transactions.

The L1 instruction cache is a 16 KB direct-mapped cache with 256-byte lines and a 20 cycle miss penalty. The L1 data cache is a 32 KB direct-mapped cache with 64-byte lines and a 15 cycle miss latency. It is non-blocking with an 8-entry miss buffer and uses a write-allocate/write-back policy with an 8-entry write buffer. The L2 cache is a 256 KB 4-way set associative cache with 64-byte lines and a 144 cycle miss latency (80 cycles attributed to DRAM access time and memory

Instruction	Latency (clocks)
ALU	1
Branches	1
Store	3
Load	4
Floating-point	4
Multiply	7
Divide	30

Table 1. Instruction latencies for processor model.

controller overhead, and 64 cycles attributed to data transfer across the bus). It is non-blocking with an 8-entry miss buffer and uses a write-allocate/write-back policy with an 8-entry write buffer. Further details available in [12].

In the new architecture, the optical technology described in Section 2 is used to provide the external path to memory, replacing the electrical bus, as illustrated in Figure 4. On the processor and memory controller chips, the original bus interface is replaced with an electro-optical interface built using 2-D arrays of VCSELs and photodetectors. A 16 × 16 VCSEL array operating at 1 Gb/s per laser yields an off-chip bandwidth of 256 Gb/s. Since the total path length is similar to that of the electrical bus, the overall latency should be comparable.

4. Evaluation Environment

This architecture evaluation uses a variety of media applications, most of which derive from the MediaBench benchmark suite defined by Lee, et al. [13]. Media benchmarks were selected for this evaluation for two reasons. First, media processing continues to dominate an ever-increasing portion of computing workloads. Second, media applications are typically characterized by enormous amounts of streaming data. Limited bandwidth may become a significant bottleneck to such applications with the growing processor-memory gap.

Among the media benchmarks are three video decoding applications and two image decoding applications. These are summarized in Table 2. Graphics and audio/speech applications are also important media applications, but graphics applications are typically off-loaded to graphics processors, and audio/speech benchmarks do not tend to be as computationally critical or data-intensive as video/image processing. For image and video, only decoding applications were evaluated in this preliminary investigation. Decoding applications are generally more memory-bound than encoding applications. Encoding applications usually perform much more processing per data element, so they are nearly always computationally-bound. Additionally, decoding applications are more common to the standard media workload since most video/image data follows the single-production/multiple-consumption use paradigm.

During the performance analysis, two separate input data sets (*input-1* and *input-2*) were used for each benchmark to identify performance variations within applications. For all benchmarks except *mpeg4dec*, the first input was that originally provided by the MediaBench developers. Table 3 gives the trace statistics for each benchmark.

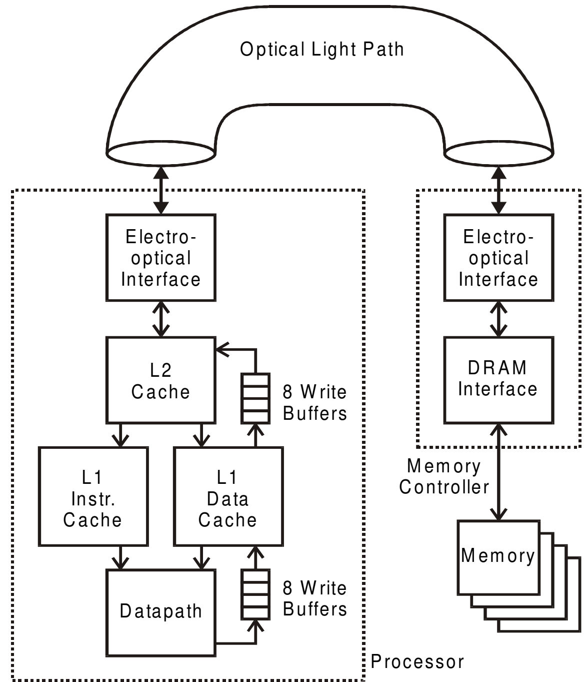

Figure 4. Computer architecture with an optical processor/memory link.

The various media applications exhibit significantly different memory performance, as displayed below in Figure 5. The first three applications spend less than 5-10% of their execution time on memory stalls for L1 and L2 misses, whereas the last two applications spend 25-30% of their time in the memory subsystem. Based on these statistics, we expect that the last two applications, *mpeg4dec* and *unepic*, will display much greater variation in performance as we exchange the electrical bus with a much higher bandwidth optical bus.

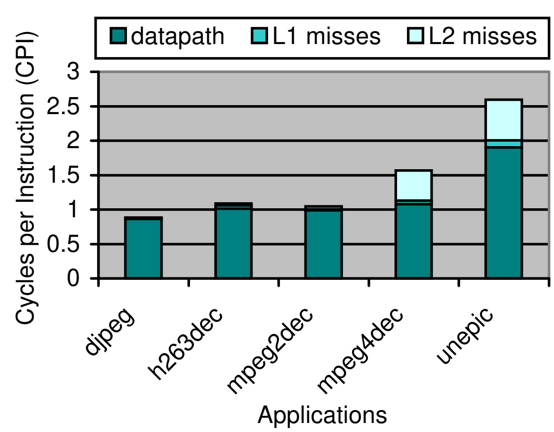

Figure 5. CPI Breakdown for *input 1* of benchmarks.

Benchmark	Description
djpeg	Lossy image compression decoder for color and gray-scale images, based on the JPEG standard; performs file I/O but no graphical display
h263dec	Very low bit-rate video decoder based on the H.263 standard; performs file I/O but no graphical display; provided by Telenor R&D
mpeg2dec	Motion video compression decoder for medium to high-quality video, based on MPEG-2 standard; performs file I/O but no graphical display
mpeg4dec	Motion video compression decoder using an object-based representation; based on the MPEG-4 standard; performs file I/O but no graphical display; provided by the European ACTS project MoMuSys
unepic	An image compression decoder based on wavelets and run-length/Huffman entropy coding

Table 2. Descriptions of applications in media benchmark.

Program	# Static Instrs	Input-1		Input-2	
		File Size	# Dynamic Instrs	File Size	# Dynamic Instrs
djpeg	19,397	5756	3M	31,074	25M
h263dec	8721	20,364	60M	19,338	65M
mpeg2dec	9520	34,906	95M	1,593,409	720M
mpeg4dec	108,273	39,213	1400M	503,060	500M
unepic	3767	7432	5M	10,129	5M

Table 3. Program and trace statistics for both input data sets.

The compilation and simulation tools for this architecture evaluation were provided by the IMPACT compiler, produced by Wen-mei Hwu's group at UIUC [14,15]. The IMPACT environment includes a trace-driven simulator and an ILP compiler. The ILP compiler supports many aggressive compiler optimizations including procedure inlining, loop unrolling, speculation, and predication. The IMPACT simulator is a parameterizable, emulation-based trace-driven simulator that enables both statistical and cycle-accurate simulation of a variety of microprocessor architecture models, including in-order superscalar, out-of-order superscalar, and VLIW data paths. The results for this initial investigation use an in-order superscalar processor model and only apply traditional compiler optimizations.

Like many common performance analysis environments [16], the IMPACT simulator employs trace sampling to avoid unreasonably long simulation times during cycle-accurate simulation of large traces. The sampling method specifies two parameters: the number of instructions in each simulation sample, and the number of instructions to skip between samples. The IMPACT developers recommend a sample size of 200,000 instructions, with the number of instructions to skip specified by the following equation:

$$max\_skip\_size = \max\left(\left[\frac{\min\left(1x10^9, trace\_size\right)}{50} - sample\_size\right], 0\right)$$

The above equation provides progressive degrees of sampling according to application size. For applications with 10M instructions, full sampling is necessary, while applications with 100M instructions and 1B instructions may require as little as 10% and 1% sampling, respectively. Sampling by these criteria is reputed to enable accuracy within 5% of that from simulating the entire trace [17]. This error range should certainly hold for multimedia applications, which have more predictable compute patterns than general-purpose applications.

It is questionable however, for what range of target architectures this accuracy holds. The IMPACT developers do not specify precise criteria regarding the acceptable range of target architectures. Other studies in trace sampling have found that sampling ratios of 10% work very well. A trace sampling evaluation by Martonosi, et. al. [18] found that sampling with a ratio of 10% and sample sizes of 0.5M instructions gave an absolute error of less than 0.3% when using smaller cache sizes (of up to 128 KB), but much larger sampling sizes are needed for cache sizes of 1MB and up. In our own simulations, we also found that accuracy degenerates on architecture simulations modeling long external memory latencies. To ensure acceptable accuracy, we doubled IMPACT's recommended sample size to 400,000 instructions, and used skip sizes of only half that specified by the above equation. Our initial investigations into the simulation accuracy of trace

Program	# Dynamic Instrs	Skip Size	% Program Simulated	Simulation Time (min)	Simulation Time (max)
djpeg	3M	0	100	0.43 min	0.48 min
h263dec	60M	1.5M	23.6	1.10 min	1.66 min
mpeg2dec	95M	2.5M	14.8	2.64 min	3.00 min
mpeg4dec	1400M	9.6M	4.0	6.20 min	9.14 min
unepic	5M	0	100	1.25 min	2.92 min

Table 4. Program and trace statistics for both input data sets.

sampling under these conditions indicate the errors from trace sampling are well within a 5% error margin. Table 4 gives the trace-sampling statistics for the benchmarks, simulated on a 933 MHz dual-processor Pentium III with 2 GB RAM.

5. Experiments and Results

We performed three experiments in comparing the performance of optical processor-to-memory buses to all-electrical buses. The first experiment examines the benefit of optical buses on current technology by starting with the base processor model and simply scaling the bandwidth up from 8 Gb/s to 256 Gb/s, without varying the latency. The second experiment evaluates the benefit of optical buses on future systems by measuring the performance variation of optical buses versus electrical buses with respect to the growing processor-to-memory latency gap. Finally, the third experiment examines the benefit of the increased optical bandwidth with respect to a simple latency-hiding technique.

5.1. Optical Bus in Current Technology

An immediate benefit can be obtained from applying optical processor-to-memory buses with today's processor technology. The significantly greater bandwidth helps decrease the long external memory latencies by virtually eliminating the data transfer time of external memory accesses. We can model the service time for an L2 cache miss with the following equation:

$$T_{L2\_miss} = T_A + \frac{T_T}{X_B}$$

Here X_B is the bandwidth factor ($X_B \in 1, 2, 4, \ldots$) with respect to the base processor's bandwidth of 8 Gb/s, T_A is the memory access time (80 ns; 70 ns DRAM access time + 10 ns memory controller overhead), and T_T is the memory transfer time (64 ns; transfer time based on a 1 GHz processor with a 64-bit bus and 8:1 processor-to-bus ratio). So each doubling of the processor-to-memory bandwidth effectively halves the data transfer time. Consequently, the bandwidth factor between an 8 Gb/s electrical bus and a 256 Gb/s optical bus is $X_B = 32$, and the transfer time drops from 64 ns to 2 ns (assuming DRAM memory sub-system provides sufficient throughput to match optical bandwidth).

In this experiment, we simulated the benchmarks on processors with bandwidth factors ranging from 1x to 8x (i.e. 8 Gb/s to 64 Gb/s)[2]. Figure 6 displays the speedups of an optical bus with 8x bandwidth factor (64 Gb/s) versus an electrical bus with a 1x bandwidth factor (8 Gb/s). The average speedup across all benchmarks is 6-7%, and benchmarks with significant memory stall penalties have performance gains in excess of 10%.

More importantly, further consideration of the results indicates an average reduction of more than 50% in the L2 miss CPI (cycles per instruction) penalty. As shown in Figure 7, the CPI for L2 misses dropped significantly with increasing bandwidth. The last two benchmarks, *mpeg4dec* and *unepic*, demonstrate CPI reductions of approximately 60% for a bandwidth factor of 8x versus 1x. And while it is difficult to discern from the figure, the other three

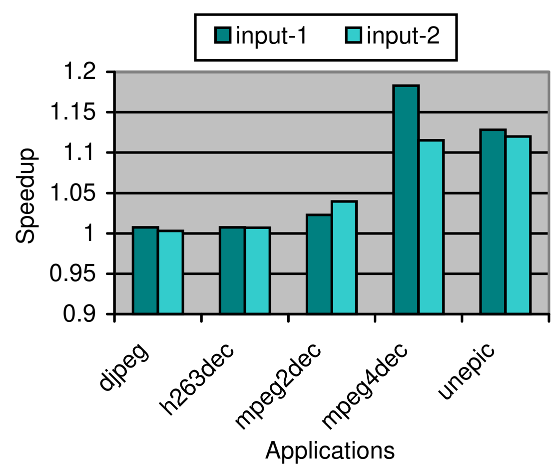

Figure 6. Speedup of optical bus vs. electrical bus for both simulation inputs.

[2] The simulator currently can only model bandwidths up to bandwidth allowed by the L2 line size (i.e. 64 Gb/s).

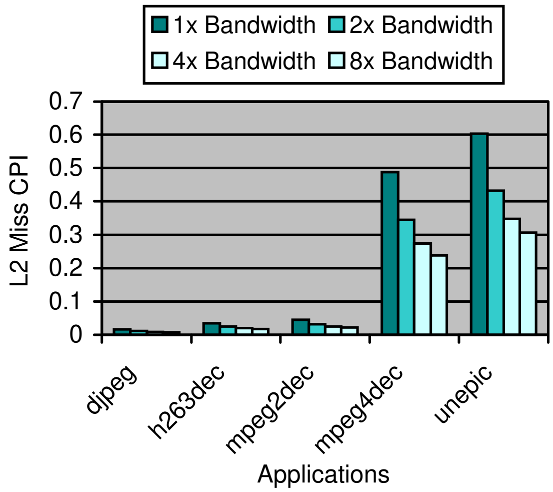

Figure 7. L2 Miss CPI for input 1 of media benchmarks.

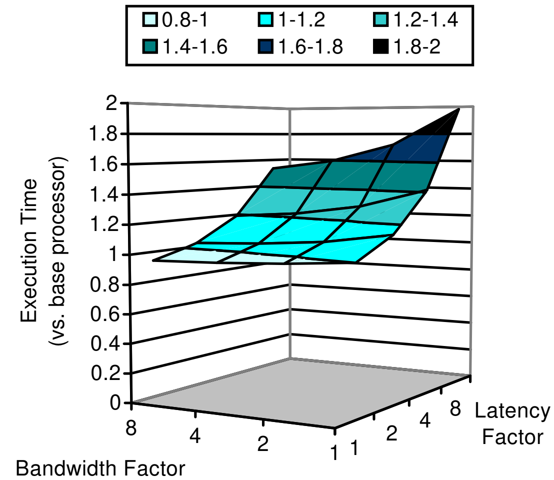

Figure 8. Average bandwidth vs. latency curve for all benchmarks on *input 1* using a processor with 64-byte L2 line size.

applications also display similar drops in L2 miss CPI. This reduction of L2 miss CPI was consistent across both input data sets, as well as for buses supporting either single or split bus transactions. Overall, we find that the increased optical bus bandwidth reduces memory stall penalties by virtually eliminating the memory transfer time.

5.2. Optical Bus in Future Processors

The first experiment demonstrated the advantage of optical buses in enabling significant L2 miss CPI reductions with current processor technology, but we also desire an understanding of the impact of optical buses in future processors. The critical trend with respect to the memory hierarchy in future processors is the continually increasing processor-to-memory latency gap. Consequently, our second experiment evaluates the effectiveness of optical buses in trading-off bandwidth for latency as we increase the processor-to-memory latency ratio.

Using the same base equation from above, we model the increasing processor-to-memory latency gap via an additional term, latency factor (X_L), to generate the modified equation:

$$T_{L2\_miss} = X_L * \left(T_A + \frac{T_T}{X_B} \right)$$

For simplicity's sake, we chose to scale both memory access time and memory transfer time by the same latency factor instead of scaling each individually. While the two times generally do not scale at the same rate, both result in increasing L2 miss times, and what is most important is the overall characteristic trend of bandwidth versus latency. There are cases in which the access time grows faster than transfer time (i.e. slower increase of memory speed vs. processor frequency than bus frequency vs. processor frequency), and conversely cases in which the transfer time grows faster than the access time (i.e. a multi-hop optical bus in which data must cross multiple links to reach its destination). Depending upon the actual memory access and memory transfer scale factors, the appropriate curve can be tracked on the overall 3D curve of latency vs. bandwidth.

Figure 8 shows the average characteristic curve of bandwidth vs. latency for the media benchmarks on *input 1*. As can be seen, the ratio of execution time with respect to the base processor increases most significantly for bandwidth factor 1x as the latency factor increases from 1x to 8x, reaching an average execution time ratio of nearly 2x (i.e. processor with X_B = 1x and X_L = 8x runs 2x slower than base processor model). However, by increasing the bandwidth from 1x to 8x when the latency factor is 8x, the execution time ratio drops to only 1.5x, which is nearly a 50% drop in memory penalties.

As in the first experiment, there was a wide variation in performance between the different media applications. Once again, the two benchmarks with significant memory stall CPIs, *mpeg4dec* and *unepic*, were more heavily affected than the other benchmarks, with execution time ratios ranging up to 3.3x. However, none of the applications demonstrated any excessive performance variations that can be attributed to limited bandwidth, even in the worst case scenario of the model with 1x bandwidth factor and 8x latency factor. The bus utilization for these applications never exceeded 50%. This can be

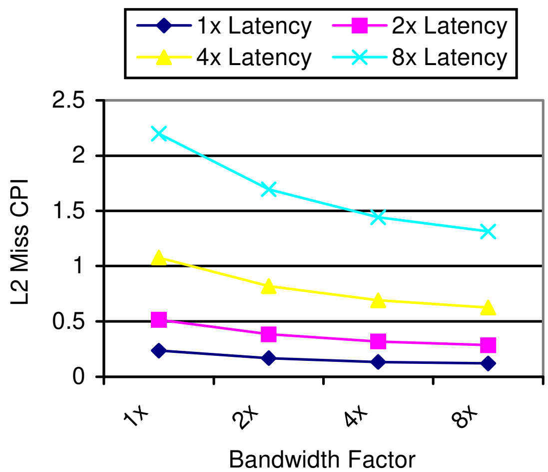

Figure 9. L2 Miss CPI for *input 1* of benchmarks.

attributed to the fact that, with the exception of *unepic*, the working set size of these benchmarks fits within the L1 data cache, and the working set size of *unepic* fits easily within the L2 cache [12].

The most important findings among these results are the following. First, the average reduction in execution time ratio between bandwidth factors 8x and 1x is approximately 50% across the curve. Second, the overall of shape of the bandwidth vs. latency curve is consistent over all the media benchmarks, and is irrespective both of input data set and bus style (single or split transaction) as well. Consequently, we find that the increased optical bus bandwidth is consistent in offering approximately 50% reduction in external memory latency penalties across a wide range of architectural variations.

The one area in which the results do show some performance variation is with respect to the latency factor. Figure 9 illustrates the average L2 miss CPIs with respect to latency factor. Overall, as the bandwidth factor increases from 1x to 8x at any given latency, the L2 miss CPI is reduced by approximately 50%. However, a closer look indicates that the reduction in L2 miss CPI slowly decreases with increasing latency factor. The L2 miss CPI reduction is 51%, 45%, 42%, and 38% for latency factors of 1x, 2x, 4x, and 8x, respectively. Consequently, we expect that L2 miss CPI reduction will continue to slowly decrease with even higher latency factors.

5.3. Optical Bus with Additional Prefetching

The final experiment attempts to evaluate the performance of utilizing the extra available bandwidth from optical buses to perform more aggressive latency hiding techniques, such as prefetching. The most basic form of prefetching is simply increasing cache line size to take advantage of spatial locality. In applications where significant spatial locality exists, this often results in increased performance. However, the increased line size may result in both increased latency since extra data transfer cycles are needed, and additional cache line conflicts since the number of cache lines is reduced. Consequently, this method may increase bandwidth.

To evaluate the impact of additional prefetching on optical buses versus electrical buses, this experiment examines the bandwidth and latency characteristics of larger L2 line sizes, such as 256- and 1024-byte lines. Figure 10 shows the average results, with the results for the electrical bus (i.e 1x bandwidth factor) on the left, and those for the optical bus (i.e. 8x bandwidth factor) on the right. In both cases, the performance improved consistently with increasing L2 line size. In neither case was the extra prefetching bandwidth sufficient to make bandwidth the bottleneck. These applications were unable to model the benefits of increased optical bandwidth in alleviating performance degradation for bandwidth-limited applications.

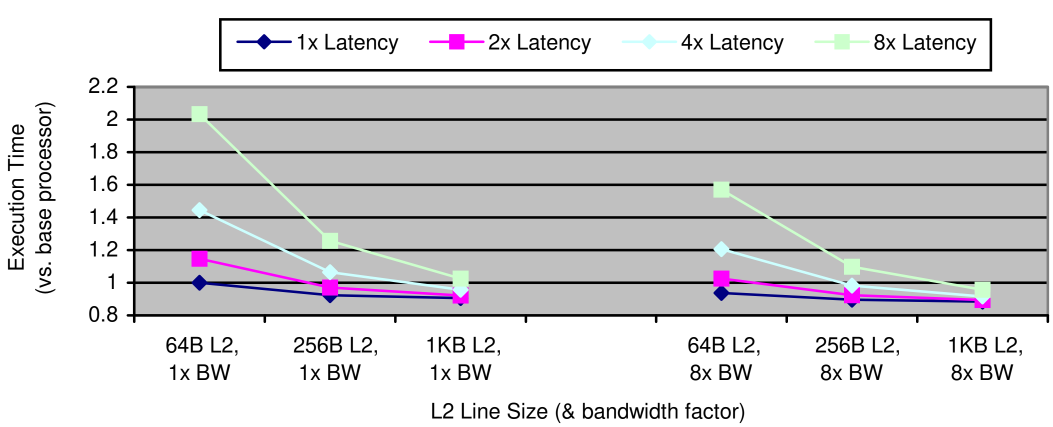

Figure 10. Average of latency and bandwidth vs. L2 line size using *input-1*.

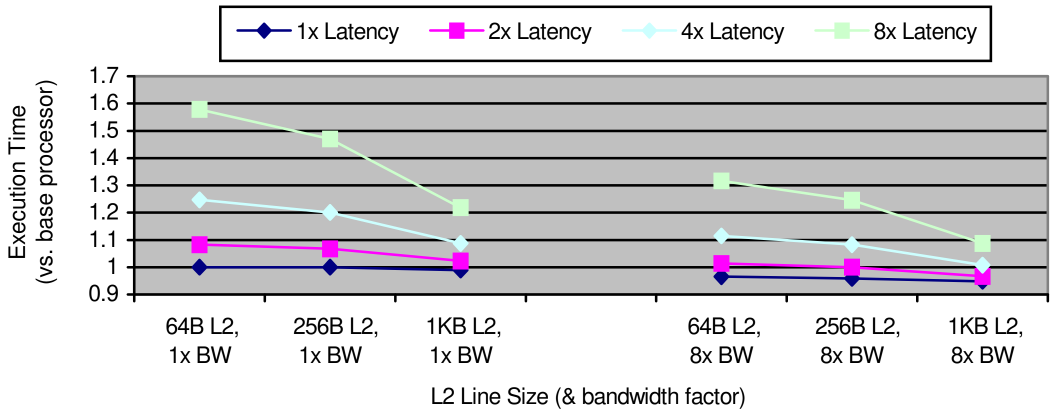

Figure 11. Diagram of latency and bandwidth vs. L2 line size for *mpeg2dec* on *input-2*.

However, the results did further validate the consistency of the increased bandwidth in reducing the impact of long(er) external memory latency penalties. Nearly all of the benchmarks showed the same characteristic curve when evaluating increased L2 line sizes. Figure 10, which shows the average execution time ratios across all benchmarks, illustrates this common characteristic curve. As L2 line size increases from 64 to 256 bytes, and then from 256 to 1024 bytes, the advantage of extra spatial locality serves to decrease execution time by approximately 70% and 60%, respectively. This is as expected, since spatial locality typically decreases with increasing distance between elements in memory. Consequently, the single effect of increasing the bandwidth from 1x up to 8x was to "compress" the characteristic curve. The resulting curves for the electrical and optical buses are identical in shape. This is true for nearly all of the media applications except *mpeg2dec*. Consequently, the impact of increased bandwidth is consistent in decreasing L2 miss CPI, irrespective of L2 line size.

The results for *mpeg2dec* were distinct from the other applications, but it also validates the consistency of increased bandwidth in reducing L2 miss CPI. As shown in Figure 11, unlike the other applications, there was only a 15% performance increase from increasing the L2 line size from 64 to 256 bytes, and then a 60% performance gain from increasing it again to 1024 bytes. Essentially, the L2 line size at 256 bytes did not prefetch data sufficiently far in advance to achieve a significant performance gain. Only by increasing the L2 line size to 1024 bytes, thereby performing even more aggressive prefetching (i.e. prefetching further into the future), did a significant gain result. Regardless, all the resulting curves for both electrical and optical buses are again identical in shape. The results indicate that across architectural variations, increased bandwidth is consistent in decreasing L2 miss CPI.

Overall, we found that increasing bandwidth via optical buses serves to reduce the impact of long external memory latencies, decreasing the L2 miss CPI an average of 50% over processors with electrical buses by effectively eliminating the data transfer time. This reduction of external memory latency penalties is shown to be consistent across many architectural variables, with only a slight reduction with increasing latency factors.

6. Further Architectural Options

This paper demonstrates just one of the many benefits of optical buses. While the significantly increased bandwidth of optical processor-to-memory buses is effective at reducing memory stall penalties by an average of 50%, the increased memory bandwidth offers several additional benefits. One obvious benefit is in eliminating the bandwidth bottleneck for bandwidth-limited applications. Other benefits include opportunities for aggressive latency-hiding methods (which often require significant extra bandwidth), data prefetching, compound buses, single-chip multiprocessors, and other novel cache and memory hierarchy designs.

Latency hiding and data prefetching is one area that may offer substantial gains from the increased bandwidth of optical buses. A variety of effective latency hiding methods exist, but they often consume considerable extra memory bandwidth. Data prefetching in particular tends to significantly increase bandwidth, often by as much as 50% or more. Research has gone into actively reducing this extra bandwidth, but the result of reducing the bandwidth limits the aggressiveness of prefetching. And with the constantly increasing processor-memory latency gap, we shall continue to need ever more aggressive prefetching to manage the latency.

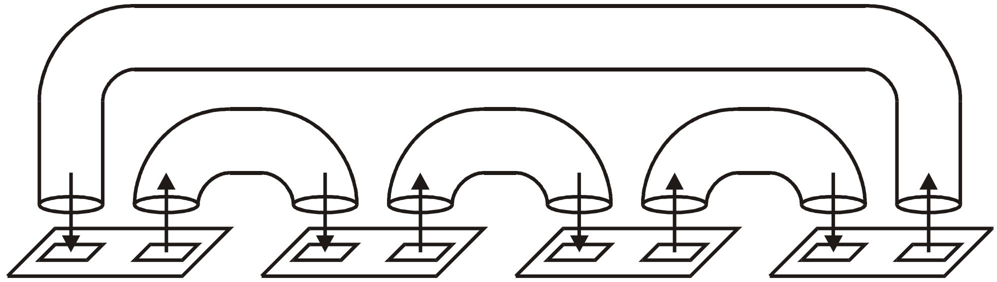

Figure 12. A four chip optical ring topology.

Single-chip multiprocessors is another field that can substantially benefit from optical buses. The conventional, multi-issue, ILP-based microprocessor architecture [19] is expected to eventually become obsolete. Increasing processor frequencies require longer execution pipelines, and thereby longer operation latencies, which impede ILP scheduling for high IPC. Consequently, wide-issue processors will no longer be able to achieve effective utilization of their functional units. The alternative is to use coarser-grained parallelism methods as provided by parallel or multi-threaded processors. It is now possible to place multiple processors on a chip, but all these processors must share the same bandwidth. This will result in a greater likelihood of applications becoming bandwidth-limited on single-chip multiprocessors. However, optical buses can be used to overcome this bandwidth limitation.

A final architectural option is a multi-hop optical bus. With optical buses enabling orders of magnitude greater bandwidth than electrical buses, it is apparent that the bandwidth of these buses will often not be fully exploited. Using a multi-hop optical bus, such as the four-point unidirectional ring optical network shown in Figure 12, an optical bus could easily support multiple peripherals and/or memory banks on a single network. The individual chips on the ring can be processors, memory controllers, or other peripheral devices. A benefit of this ring topology is that standard cache coherence mechanisms will function properly provided transactions are propagated all the way around the ring.

These are just a few of the architectural options that are enabled by optical buses. The extraordinary bandwidth of optical buses offers limitless opportunities.

7. Conclusions

Optical technology has long been an effective method for communications and interconnects between components separated by distances on the order of meters to kilometers. Now, new electro-optical technology has begun to enable the use of optics in connecting devices on a much smaller scale. This technology has been previously proposed for multiprocessor interconnects, and we now propose using it as the processor-to-memory data path in microprocessor systems. Such optical buses will enable orders of magnitude greater bandwidth between the processor and off-chip memory with no appreciable latency penalty.

This paper provides a preliminary evaluation of the benefits of an optical processor-to-memory bus in both eliminating the bandwidth bottleneck and in reducing the impact of the increasing processor-to-memory latency gap. We assess the performance impact of this architecture enhancement on a number of media applications, and examine its benefit both with respect to current processor technology, and for use with future processors. Overall we found that the increased bandwidth nearly eliminates the transfer time between processor and memory, effectively reducing penalties from long off-chip memory latencies by 50% on average. Furthermore, we found that this reduction of the L2 miss CPI is consistent across a wide-range of architectural variations, decreasing only slightly with increasing memory latency. Finally, significant additional bandwidth remains, opening the door to many advanced architectural features, including aggressive latency hiding techniques, single-chip multiprocessors, and multi-hop optical buses. The orders of magnitude extra bandwidth provides extraordinary opportunities for advanced architecture research in microprocessor systems.

[1] D. Burger, J.R. Goodman, and A. Kagi, "Limited bandwidth to affect processor design," *IEEE Micro*, 17(6):55-62, 1997.

[2] Y. Li, E. Towe, and M. Haney, eds., *Proceedings on Short Distance Optical Interconnections in Digital Systems*, 2000.

[3] D. Plant et al., "A 256 channel bi-directional optical interconnect using VCSELs and

photodiodes on CMOS," *Proceedings of Optics in Computing*, pp. 1046-1054, June 2000.

[4] M. Chateauneuf et al., "Design, implementation and characterization of a 2-D bi-directional free-space optical link," *Proceedings of Optics in Computing*, pp. 530-538, June 2000.

[5] H. Kosaka et al., "A two-dimensional optical parallel transmission using a vertical-cavity surface emitting laser array module and an image fiber," *IEEE Photonics Technical Letters*, 9:253-255, 1997.

[6] T. Maj et al., "Interconnection of a two-dimensional array of vertical cavity surface emitting lasers to a receiver array via a fiber image guide," *Applied Optics*, 39(5):683-689, 2000.

[7] Ch'ng Shi Baw, R. Chamberlain, and M.A. Franklin, "Design of an interconnection network using VLSI photonics and free-space optical technologies," *Proceedings of the 6th International Conference on Parallel Interconnects*, pp. 52-61, October 1999.

[8] R. Chamberlain, M.A. Franklin, and A. Mahajan, "VLSI photonic ring interconnect for embedded multicomputers: Architecture and performance," *Proceedings of the 14th Conference on Parallel and Distributed Computing Systems*, August 2001.

[9] A. Mahajan, M.A. Franklin, and R. Chamberlain, "Fairness issues in an embedded photonic ring interconnect," *Proceedings of the 4th High Performance Embedded Computing Workshop*, September 2000.

[10] R. Chamberlain, M.A. Franklin, and P. Krishnamurthy, "Performance evaluation of a reconfigurable, embedded photonic multi-ring interconnection network," *Proceedings of the 5th High Performance Embedded Computing Workshop*, November 2001.

[11] L. Gwennap, "Digital 21264 sets new standard," *Microprocessor Report*, 10(14), Oct. 1996.

[12] J. Fritts, "Architecture and compiler design issues in programmable media processors," Ph.D. Thesis, Department of Electrical Engineering, Princeton University, 2000.

[13] C. Lee, M. Potkonjak, and W.H. Mangione-Smith, "MediaBench: A Tool for Evaluating and Synthesizing Multimedia Communications Systems," *Proceedings of the 30th Annual International Symposium on Microarchitecture*, December 1997.

[14] P. P. Chang, S. A. Mahlke, W. Y. Chen, N. J. Warter, and W. W. Hwu, "IMPACT: an architectural framework for multiple-instruction-issue processors," *Proceedings of the 18th Annual International Symposium on Computer Architecture*, Toronto, Canada, May 1991.

[15] IMPACT compiler group at UIUC: http://www.crhc.uiuc.edu/Impact

[16] P. Bose and T.M. Conte, "Performance Analysis and Its Impact on Design," *IEEE Computer*, 31(5):41-49, May 1998.

[17] Personal conversations with John Gyllenhaal and Brian Deitrich.

[18] M. Martonosi, A. Gupta, and T. Anderson, "Effectiveness of Trace Sampling for Performance Debugging Tools," *Proceedings of the 1993 ACM SIGMETRICS Conference on Measurement and Modeling of Computer Systems*, 1993.

[19] V. Agarwal, M.S. Hrishikesh, S. W. Keckler, and D. Burger, "Clock Rate versus IPC: The End of the Road for Conventional Microarchitectures," *Proceedings of the 27th Annual International Symposium on Computer Architecture*, 2000.