

Parallel optical interconnects may reduce the communication bottleneck in symmetric multiprocessors

Jacques Henri Collet, Wissam Hlayhel, and Daniel Litaize

We start from a detailed analysis of the communication issues in today's symmetric multiprocessor (SMP) architectures to study the benefits of implementing optical interconnects (OI) in these machines. We show that the transmission of block addresses is the most critical communication bottleneck of future large SMP's owing to the need to preserve the coherence of data duplicated in caches. An address transmission bandwidth as high as 200-300 Gb/s will be necessary in typically ten years from now; this requirement represents a difficult challenge for shared electric busses. In this context, we suggest the introduction of simple point-to-point OI's for a SMP cache-coherent switch, i.e., for a VLSI switch that would emulate the shared-bus function. The operation might require as much as 10,000 input-outputs (IOs) to connect hundred of processors, particularly if one maintains the present parallelism of transmissions to preserve a large bandwidth and a short memory access latency. The interest for OI's comes from the potential increase of the transmission frequency and from the possible integration of such a high density of IO's on top of electronic chips to overcome packaging issues. Then, we consider the implementation of an optical bus that is a multipoint optical line involving more optical technology. This solution allows multiple simultaneous accesses to the bus, but the preservation of the coherence of caches can no more be maintained with the usual fast snooping protocols.

OCIS Codes: 200.4650, 200.2610

I. Introduction

Today, all general-purpose microprocessors are designed to operate with a hierarchy of caches L1, L2 (and sometimes L3) to limit the frequency of the accesses to the main-memory [1]. The role of caches is essential because the time needed by the processor to fetch instructions (or data) from the main memory has increased permanently (in terms of processor cycle), making the access latency to the main memory more and more critical for the *global performance* of the system. We recently discussed the latency issue in mono and multiprocessors in relation with the possible implementation of optical interconnects [2]. Concretely, the access time to L1 costs 1 (processor) cycle, that to L2 typically 4-5 cycles, and accessing the main memory

requires from 40 to 80 processor cycles. With the hierarchy of caches, the average access time to data (or instruction) reduces in a monoprocessor to 2-4 cycles (depending of the cache miss rate) that must be compared to the 40-80 cycles just mentioned.

The utilization of caches in a general-purpose multiprocessor environment is still more critical because the memory access time is increased by the appearance of several additional latencies, namely:

- 1) A latency to access the interprocessor network. Even in a network as simple as a shared-bus, the access arbitration may last a few processor cycles, depending on the number of processors and on the shared bus bandwidth.
- 2) A latency owing to the propagation through the network, depending on the internode distance. Node processing latency dominates in tightly bound multiprocessors. The constraint of propagation latency exists even in a shared-bus machine as will be discussed in section III.
- 3) Another latency to maintain the coherence of the memory hierarchy. The *coherence issue* comes from the fact that several copies of the same data can exist with different values in different caches and in the main memory, so that each processor must know, when attempting to access duplicated data, which

J.H. Collet (Collet@laas.fr) is with the Laboratoire d'Analyse et d'Architecture des Systèmes du Centre National de la Recherche Scientifique, 7 av. du colonel Roche, F-31077 Toulouse CEDEX, France. Wissam Hlayhel and D. Litaize are with the Institut de Recherche en Informatique, Université Paul Sabatier; 118 route de Narbonne F-31062 Toulouse France.

Received 26 April 2000; revised manuscript received 12 February 2001

copy is valid and which one(s) is (are) obsolete. The fastest method to maintain the coherence consists of implementing electronic circuits that execute the coherence protocols. One may distinguish:

- Snooping protocols, which only work if all processors can watch the transactions with the memory at a single synchronization point (as in shared buses), thus allowing simultaneous update or invalidation of duplicated copies of data in all caches [3].
- Directory-based protocols that work in any kind of network topology [4].

Propagating data blocks through the network and maintaining the coherence of caches in a multiprocessor environment costs time. The bus is the simplest network and snooping protocols are generally much faster than directory-based protocols because broadcasting in bus architectures occurs on the bus length, i.e., on a short distance. Thus, bus-based machines [i.e., symmetric multiprocessors (SMP's)] that use a snooping protocol generally show a lower memory access latency (MAL), are efficient and preferable, *provided that they can be constructed*. Here is in fact the heart of the matter. The feasibility of large SMPs depends on the capability to reach the huge bandwidth needed at the serialization point (i.e., the bus). We analyze the advantages of introducing optical interconnects (OIs), particularly to solve the address transmission issue. Although the transfer of data requires much more bandwidth, this transfer is less critical as the bandwidth of data links can be increased with no coherence constraint. The rest of the work is organized as follows:

- 1) We describe the current state of SMPs, particularly the communication performance.
- 2) Using a multiprocessor simulator, we study the address bandwidth requirements of the next generation of SMP's. This work shows that an address-transaction bandwidth as high as 200-300 Gbit/s will be necessary to not significantly slow down the execution of most applications in future large SMP's (including for instance 64 superscalar processors operating at a few GHz). Reaching such a high bandwidth is particularly difficult using electric shared buses because of their intrinsic operation constraints. It seems necessary to search for other implementations of the address transmission network.
- 3) We consider the introduction of OI's. We review two solutions in the ascending order of complexity of the optical subsystems, as one critical issue concerns the degree of sophistication, the feasibility and the cost of the optical parts. First, we consider simple point to point OI's for a SMP cache-coherent switch, i.e., for a switch that emulates the shared-bus function. The operation of such a chip might require as much as 10,000 input-outputs (IOs) to connect a hundred of processors, particularly if one maintains the parallelism of transmissions to preserve low MAL. The interest for OIs comes from the huge

transmission frequency and from the packaging simplification achieved by the possible integration of thousands of IOs on top of electronic chips. Then we consider the introduction of an optical folded bus operating in the synchronous mode with no access arbitration that permits simultaneous transfer of different transactions to the bus. The folded bus provides a huge bandwidth, but the price to pay is high, as the coherence of caches cannot be preserved by means of the usual snooping protocols. New cache coherence protocols are under investigation.

II. Communication architecture of present Symmetric Multiprocessors

The fast increase of the processor power has induced a dramatic growth of the bandwidth needed in recent SMPs as can be shown from the evolution of shared-bus machines. For instance, the *XDBus* from Sun [5] is a 64-bit multiplexed address-data bus running at 40 MHz (10 slots), based on a split-transaction protocol interleaving the transfer of the block address and the related data. Each bus transaction requires 2 bus cycles for an address transfer and 9 bus cycles for a cache block transfer. The raw bandwidth is 320 MB/s, but the effective bandwidth is only $\frac{3}{4}$ of this value. In the SGI Power Challenge [6], the split-transaction protocol is enhanced to take into account the presence of two busses clocked at 47.6 MHz (13 slots), one of 40 bits dedicated to the address transfer and another one of 256 bits used for the transfer of cache blocks. The two busses are synchronized and enable to transfer the address and the data in 5 bus cycles. The busses can support up to 8 outstanding read requests. The raw bandwidth is 1.5 GB/s but the effective bandwidth reduces to $\frac{4}{5}$ of this value.

The *Sun Enterprise 6000* [7] is also a non-multiplexed split-bus running at 83.5 MHz with 256 bits for the data and 41 bits for the address. This high operation frequency (for a shared bus) is due to a special backplane design (so-called "Centerplane" technology) where cards are connected by both sides (8 slots per side). The raw bandwidth is 2,6 GB/s, with up to 112 requests possibly in simultaneous progress and 7 from each board. Each processing board includes two processors. In recent SMP designs, such as the *G30* machine from IBM or the *Enterprise 10000* from SUN [5], data are transmitted to the main memory through a crossbar, whereas addresses are still transferred through a shared bus to maintain the cache coherence with a snooping protocol. Above a certain number of processing elements, multiple address busses are needed (see Fig. 1). For example, two *XDbusses* were implemented in the *SparcCenter 2000* to connect 30 processors, four busses in the *Enterprise 10000* to reach 64 processors. With a smart design of the crossbar switch, the direct transfer of blocks from cache to cache becomes possible without updating the main memory. The most critical problem at the moment is the address/snoop bandwidth discussed in the next section.

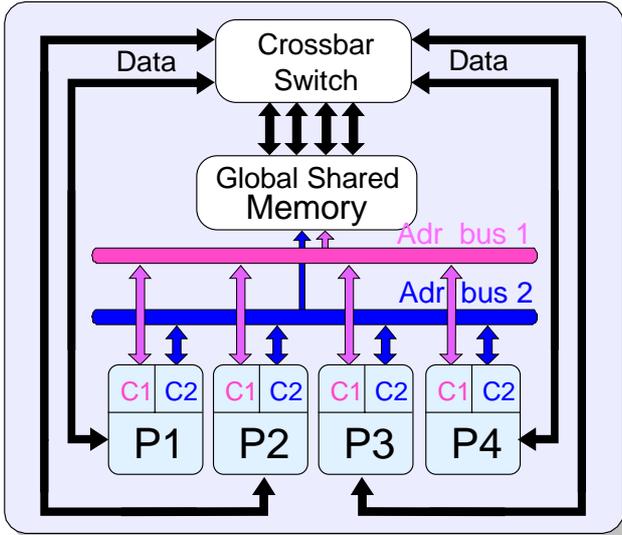


Figure 1: Example of a quadriprocessor SMP architecture with two address busses (Adr bus 1, Adr bus2) and the associated processor caches (C1 for Adr bus 1, C2 for Adr bus 2).

III. Address traffic and limitations of electric busses

We studied the dependence of the execution time of standard multiprocessor applications versus the bus bandwidth, using an instruction-driven simulator that enables us to trace the bus traffic at run time. The main features of superscalar processors (i.e., multiple instruction issue, dynamic scheduling, non-blocking reads, register renaming, branch prediction, and speculative execution) are parameters that can be adjusted to study the SMP performance. Reference 8 describes in details the simulator and previous simulation results, showing in particular the address traffic traces, which reveal the high irregularity of communications. It is important to stress that the processor cycle is the time unit of these simulations. All access times to the different elements of the memory hierarchy are expressed in processor cycle unit. We assumed that the memory is contention-free to fully characterize the impact of the bus bandwidth. The MAL is an adjustable parameter. In Fig. 2a, the MAL equals 17 cycles, which is practically equivalent to considering slow processors operating around 200-300 MHz. In Fig 2b, the MAL is increased up to 75 processor cycles, corresponding to future processors operating around 3-4 GHz. Figures 2a and 2b display the dependence of the execution time of several applications (Radix, Ocean, LU, FFT) of the standard SPLASH suite [9] versus the bus bandwidth. All execution times converge to minimal values at high bus bandwidth, when the bus congestion latency becomes small compared to the intrinsic MAL that remains the ultimate bottleneck as in the monoprocessor architecture.

Let us consider Fig. 2b which describes the behavior of future large SMP's. The execution time of most applications (LU, FFT, ocean) increases of the order of 30% when the address bus bandwidth is reduced from 1 to 0.5 request per processor cycle (RPC). The worst

case is that of the *radix* application, with an execution time approximately multiplied by 2. We may consider that 0.5 RPC is a reasonable bus bandwidth compromise, not to slow significantly the execution speed of applications.

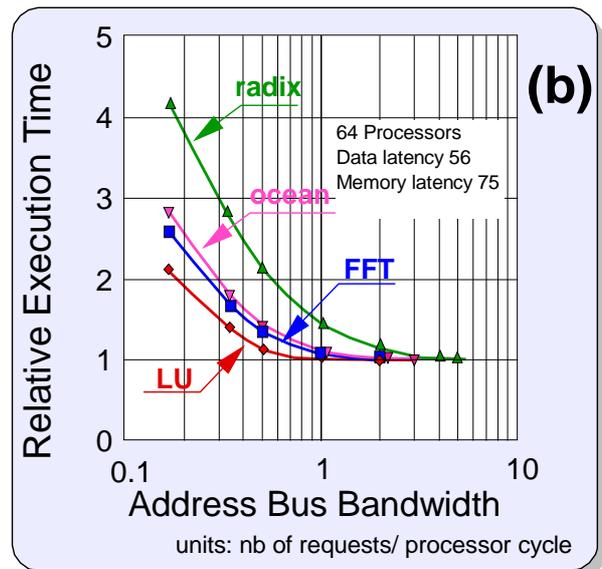
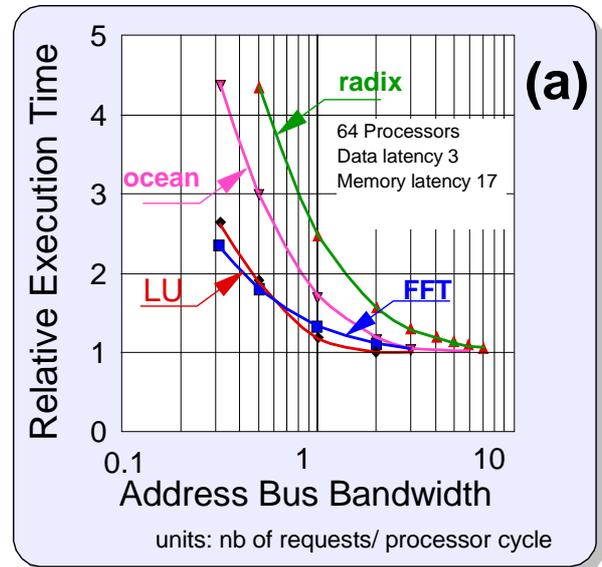


Figure 2: Execution time versus bus bandwidth for different applications of the SPLASH suite. The execution times of the different applications are normalized to 1 when considering an infinite bus bandwidth. Data and Memory latencies are given in processor clock cycle unit

Let us consider an example for concreteness, with processors operating at $F=4$ GHz (in accordance with the latency parameters of Fig 2b) and 40-bit memory addressing. As from 2 to 5 bus cycles are necessary to transmit an address (owing to the coherence controls and to whether the bus arbitration mechanism is pipelined), the requested bus bandwidth will be at least of the order of $[4 \text{ (GHz)} \times 40 \text{ (bit)} \times 0.5 \text{ (RPC)} \times 2] = 160 \text{ Gb/s}$. In that estimation, we assumed optimistically that transmitting a request requires 2 bus cycles. With processors possibly operating up to 10 GHz in ten years

from now, address transmission needs might range from 200 to 300 Gb/s. Reaching this bandwidth is possible by expanding the bus width or (and) its operation frequency. For instance, a shared bus as large as 256 bits was implemented in the Power Challenge [6], although it operated at the low frequency of 47.6 MHz. A 256-bit bus operating at 1 GHz would provide the requested bandwidth. However, increasing the operation frequency of a shared bus is (and has always been) a very difficult task due to fundamental electric constraints. Remember that a shared bus operates as a transmission line. Two points are important:

- The bus operation frequency F_B must be low enough to ensure that the bus has time to attain an equipotential state (despite propagation effects) during the bus cycle. This condition is mandatory to warrant that all processors can snoop all memory transactions every bus cycle. The operation condition is

$$d < \nu / F_B \quad \text{Eq. 1}$$

where ν is the velocity of the electric signal along the bus and d the bus length.

- The propagation velocity of logical signals along the electric bus is

$$\nu = 1 / \sqrt{LC} \quad \text{Eq. 2}$$

with L being the linear inductance (typ. 5-10 nH/cm) and C the linear capacitance. The critical point is that the average linear capacitance depends on the number of transceivers that connect the different processors, each transceiver contributing typically for $C_T=10$ -20 pF. The average linear capacitance when connecting N processors over the length d is

$$C = NC_T / d \quad \text{Eq. 3}$$

Using Eqs 1-3, we deduce after two lines of algebra:

$$d < \frac{1}{F_B^2 NLC_T} \quad \text{Eq. 4}$$

This formula leads to severe limitations. Let us consider for instance a bus operating at $F_B=100$ MHz, with $L=5$ nH/cm, $C_T=10$ pF/transceiver, which are today's parameters of the Gunning Transceiver Logic (GTL) technology [10, 11]. With $N=64$ boards, the top bus length is $d=30$ cm. At first sight, this length is relatively long, but the interboard distance would be approximately 5mm, leading to evident issues. Operation would become extremely problematical at higher operation frequency. At 1 GHz, the top length would reduce to $d \approx 3$ millimeters, due to the frequency dependence of the bus length as $1/F_B^2$ (see Eq. 4). This length seems quasi-incompatible with the constraints of processor energy dissipation and system integration. There are two known strategies to circumvent this issue. They consist of:

1. Duplicating the number of address busses (for instance considering 10 or 20 busses) to keep low the operation frequency while increasing the total bandwidth. From our previous example, it is clear that 10 busses operating at 100 MHz exhibit the same bandwidth as a single one operating at 1 GHz, without the dramatic constraint of bus length reduction. However, the price to pay is a complication of the system design (because each processor will have to communicate with 10 cache hierarchies, one per bus) and a bad management of the memory space (because each bus addresses a fraction of the memory space).

2. Limiting the number of cards by connecting multiprocessor cards.

A typical example of this evolution of the bus system consists of considering ten 64-bit busses operating at 500 MHz, connecting 16 quadriprocessor boards (i.e., 64 processors). The total bandwidth would be 320 Gb/s, in accordance with the bandwidth deduced from simulations. The top bus length would be of the order of 5 cm. It is possible that the line parameters L and C_T diminish in the future. A reduction of 50% of these two parameters would induce an extension to 20 cm of the bus length, corresponding to an interboard distance close to 1 cm with 16 boards. Therefore, it cannot be claimed that the inflation of the bus number, nor the small interboard distance, nor some additional energy dissipation issues are insuperable obstacles that will prevent the realization of fully electronic SMP's busses. However, a technological break seems necessary not to be permanently confronted to technological limits. We discuss in the following two strategies for introducing OI's in SMP's.

IV. Implementation of Optical communications in Symmetric Multiprocessors

We review the solutions, in the ascending order of complexity of the optical subsystems, on the basis of the critical issues of the solutions degree of sophistication and ultimately their cost.

A. Optical Interconnects for a Symmetric Multiprocessor Cache-Coherent Switch

The main limitation of shared electric busses comes from the capacitance of connected transceivers that slows down the propagation of electric signals, limits the operation frequency, and forces shorter bus lengths. Therefore, an efficient strategy to circumvent this limitation might consist of integrating the bus (or the different busses in the case of many-bus architectures, as shown in Fig. 1) in a single VLSI chip to reduce all dimensions and to relax the electric constraints, particularly those resulting from the "equipotentiality" condition. Note that *integrating* the bus does not really mean maintaining the bus architecture in the new chip (particularly the presence of long parallel electric lines), but rather constructing a logical circuit that emulates the bus operation, mainly the mechanism of arbitration, the serialization of the address transactions with the

memory, and the preservation of the snooping mechanism. Replacing the address bus by a cache-coherent switch (CCS) is nothing but the continuation of an idea already put into practice in the monoprocessor architecture with the so-called chipset integration. However, its generalization to multiprocessors is not trivial for several technological reasons. The most important obstacle is the chip connectivity. Connecting N processors to the CCS would require at least $64 \cdot N$ pins (maintaining today's 64-bit wide transmissions), plus $64 \cdot M$ extra connections to M memory chips, plus some more pins for the power supply, the ground, etc.. With $N=64$ processors, $M=4$ memory chips, the number of pins would be around $N_p=4500$. This estimation does not include the data transfer network (i.e., the private links to the crossbar, see Fig. 1) so that integrating the data and the address networks might require of the order of 10000 pins. The mechanical feasibility of such a high number of pins does not seem so far not demonstrated.

OI's have the potential to overcome this problem. There are many technological arguments to suggest their introduction [12,13]. Perhaps, the most important arguments are that:

- Transmitting a few Gb/s over more than several centimeters is much simpler with OI's than with electric lines. There is no line adaptation issues, no capacitance effects to slow down the propagation velocity. Remember that the propagation time equals 5 ns/m in the optical fibers that is much less than the value of 25-30 ns/m deduced for a GTL bus with one transceiver connected every second cm (see Eq. 2). Thus, an optical bus (that is of course an optical system much more complicated than a simple point to

Several arrays of photodetectors and emitters (ranging from 8x8 to 32x32) have been demonstrated by different groups with a device pitch as small as 125 microns [14, 15, 16, 17] showing that integrating a density of about 5000 optical IO/cm² on top of CMOS chips is feasible.

Therefore, OI's represent a technological breakthrough that could enable integrating the thousands of IO's necessary in the CCS approach. Figure 3 shows a possible implementation of a SMP optoelectronic chipset, particularly the presence of arrays of emitters and receivers on top of the chip. There are several solutions to interconnect the processors to the CCS. One may consider free space interconnects [18], guided transmissions through an optical backplane [19] or ribbons of fibers. All solutions seem viable. However, the simplicity of implementation, the mechanical stability (that is a critical feature of optical systems), and above all, the cost will be the final decisive arguments.

The following points are worthy of note:

- The energy consumption of the input-outputs (IO) is not a real limitation of the CCS. As shown in section III, the address bandwidth needed by a 64-processor SMP would be of the order of 200-300 Gb/s, and about ten times more to transfer data. Thus, a global bandwidth as high as 2-3 Tb/s would be necessary between the CCS, the processors, and the memory banks. Assuming a typical IO switch energy E of 20 pJ/bit ($E=CV^2$ with 1-volt bias, and 20 pF of IO capacitance), the total dissipated IO power would be of the order of 40-60 W. This estimation shows that the IO energy dissipation is not a dramatic issue to justify the introduction of OI's. Moreover, it is not clear that the replacement of electric by optical interconnects could enable reducing this consumption [12, 13, 20]. As stressed in Ref. 20, the fundamental problem is to compare the signal power requirements of optical an electrical links under the constraints of *equal bandwidth and signal-to-noise ratio*.

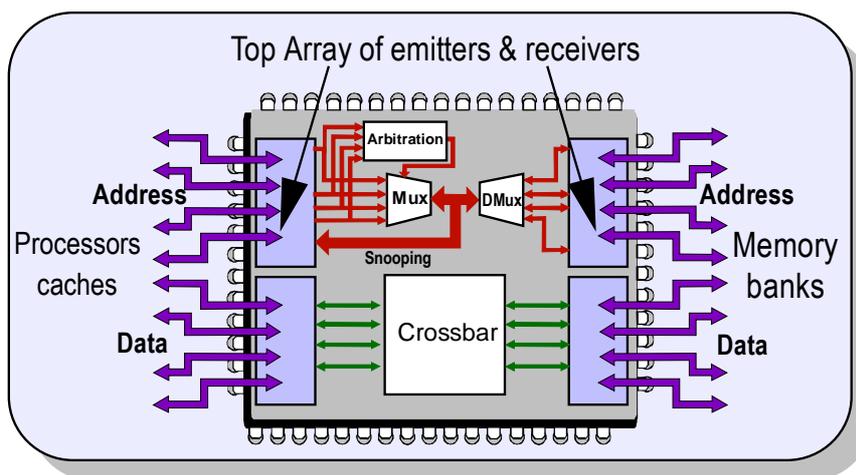


Figure 3: New SMP implementation including a central multiprocessor chipset with optical links connecting the chipset to the processors and the memory banks.

point optical link) could *a priori* operate at higher frequency than an electric one.

- The vertical emission of optoelectronic devices such as VCSEL's enables integrating optical IO's anywhere on top of VLSI chips, not only on the perimeter.

do not need to absolutely operate at very high frequencies (i.e., at a few Gb/s or higher). Let us consider for instance that the CCS must transmit 0.5 RPC for a 64-processor SMP as suggested in

section 3. Each processor therefore transmits an average bandwidth of 1/128 RPC (assuming traffic equipartition). Of course, a P-CCS bandwidth of 1/128 RPC would be too low, as it would imply that the transfer of one request would last 128 processor cycles, resulting in transmission latency longer than that of the memory, which is of the order of 40-80 cycles. A transmission bandwidth larger than the average value is necessary. For instance, with 1/5 RPC, the transmission would last 5 cycles. With processors operating at $F=3$ GHz and forty-bit memory addressing ($N_b=40$), the required bandwidth would be $F*N_b/5=24$ Gb/s, easily achieved with low cost electronic circuits and optoelectronic interfaces. One may consider implementing for example 40 parallel links at 600 MHz. The use of 4 links operating at 6 Gb/s is possible, but not mandatory. The cost will be likely the ultimate argument to decide which is the best compromise between parallelism of transmission and operation frequency.

B. Optical bus

The CCS approach suggested in the previous paragraph only requires point-to-point optical interconnects. It is based on the underlying idea of extending the bus bandwidth by electronic integration without questioning the basic operation principle of the system, in particular the serialization of the address transactions to the memory to enable snooping. Another alternative to circumvent the address bandwidth bottleneck in future SMP's might consist of relaxing the bus access arbitration. Two options are possible:

- Considering this approach in the framework of the CCS with a new architecture that would enable simultaneous accesses. This is electronic design.
- Implementing an optical folded bus (also designed as U-bus) as shown in Fig. 4 [21,22]. Of course, the construction of an optical bus (on-chip or off-chip) implies an increase of the involvement of the optical technologies with respect to the CCS approach described in section 4.1. The bus is a multipoint optical line. In particular, it requires that each node adds and drops optical data with concomitant energy loss and energy balance issues.

Let us describe the bus operation. The processors insert optical packets in the *LOAD Zone*, which extends from A to B in Fig. 4. The U-bus may operate in the synchronous or in the asynchronous mode. A real difficulty with the asynchronous operation is that each processor must arbitrate quickly (on the fly) when it attempts to access the bus with the priority always to the packets already in the bus at it is impossible to stop an optical packet. The synchronous operation based on the alternation of two phases is simpler. We distinguish:

- *The LOAD phase*. Each processor inserts (if necessary) one optical pulse in the bus *LOAD zone* (see Fig. 4). Although the pulse duration depends on the optical emitter technology, duration as short as a few tens of picoseconds is easily achieved today with mode-locked lasers or fibers. The only geometric restriction to avoid access arbitration during the load phase is to use optical pulses shorter than the internode distance L (i.e., $L > cW_p$ where W_p is the duration of the pulse c the velocity of the light in the transmission medium). In that case, there is no overlap of data from adjacent processors. Considering a bus as large as the address size makes possible transmitting one full address per cycle.

- *The propagation phase*. It follows the LOAD phase and lasts until the address inserted by the first processor P1 bypasses the point B. This time T_p is necessary to free completely the *LOAD zone* from data. T_p depends on the length of the *LOAD zone*, and can be as short as NW_p , where N is the number of connected processors. Thus, an optical bus may potentially transmit one address to the memory every W_p seconds. For instance with 10-ps pulses and $N=100$ processors, the bus access latency could be as short as 1 ns and the bandwidth as high as 100 GigaAddress/s. This extremely huge bandwidth is too large for existing memories, but it demonstrates the optical bus potentiality. In practice, the solution to match the bus bandwidth with that of the memory consists of moving away the nodes to slow down the bus traffic (also increasing the bus access latency). For

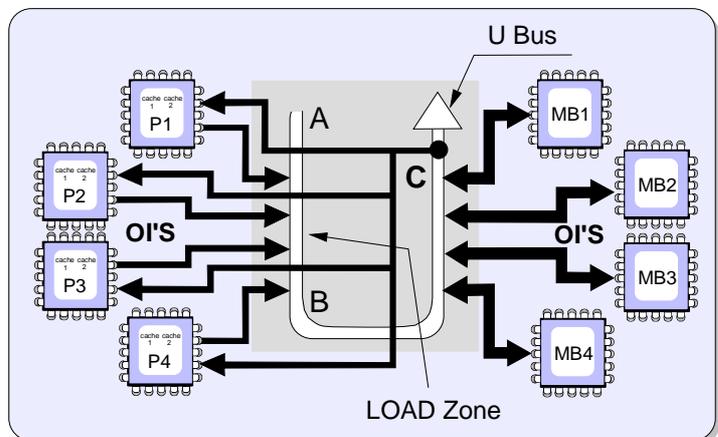


Figure 4: Example of an Optical folded bus (U-bus) connecting 4 processors to 4 memory chips. The bus width is not drawn for simplicity. A node can be a mono or a multiprocessor card. Packets are added in the *LOAD zone* extending from A to B and move towards memory chips

instance, with one node every 20 cm, the U-bus bandwidth reduces to 1 GAddress/s.

Bus operation up to a few GHz (or higher) becomes possible (in replacement of shared electric bus operating around a few tens of MHz) because the transmission of optical pulses in guides or free space is not penalized by capacitance effects and critical load adaptations encountered for electrical transmissions in a multi-point line. Moreover, the parallel transmission through optical

signals is almost skew-free in the GHz domain for transmission over a few tens of meters. This simplifies data sampling in case of parallel transmissions. As a result, the SMP architecture (i.e., the processor, the bus and the memory) would become more scalable.

Dropping the access arbitration to the U-bus simplifies the implementation and extends the bandwidth, unfortunately with the very negative side effect that cache coherence can no more be preserved with a standard snooping protocol [1]. As maintaining the coherence of caches is mandatory to avoid quickly degrading the performance of a general-purpose machine (see the discussion in the introduction), one may consider the implementation of a directory-based coherence protocol (DBCP) [1]. However, DBCP's are much slower than snooping protocols, and it is not clear that the disappearance of the access arbitration achieved by the U-bus architecture will offset the slowing down induced by a directory-based protocol. To our knowledge, the definition and the implementation of a fast coherence protocol is an open problem never discussed in the U-bus architecture. We are currently studying a new snooping protocol that is possible because the U-bus serializes all accesses to the memory, exactly at point C in figure 4. However, the atomicity of transactions is not preserved that complicates the actions to maintain the coherence of caches.

V. Conclusion

We showed that the address transmission bottleneck is the most critical communication issue of future large SMP's owing to the need to preserve the coherence of the memory hierarchy. Address bandwidths as high as 200-300 Gb/s will be necessary representing a difficult challenge for shared electric buses due to their strict operation constraints (section III). The integration of OI's could help in designing new SMP's. We considered two possible solutions, namely:

- The CCS approach, which only needs point to point OI's and that is based on two underlying ideas: 1) Including the minimum amount of optical technology, due to the economic risk factor; 2) Extending the bandwidth by integrating the electronic bus in a single VLSI chip without questioning the basic operation principle of the system, in particular the serialization of the address transactions to the memory and the preservation of the coherence with a snooping protocol. The interest of OI's comes from the potential integration of several thousands of optical IO's on top of the CCS chip to overcome packaging issues.
- The optical-bus approach based on the construction of a multipoint optical line. Optical solutions using free-space interconnects, guided transmissions through an optical backplane or ribbons of fibers are conceivable, although it is clear that these technologies are at the moment expensive and in preliminary development

stage. The cost will be the final decisive argument. The most critical question for the viability of the U-bus approach is the development of a fast protocol to preserve the cache coherence that is mandatory to maintain the SMP performance.

This research was conducted in the framework of the Optical Interconnects for Integrated Circuits project (OIIC) supported by the MicroElectronic Advanced Research Initiative (Mel ARI) of the European Communities.

References

- [1] J.L. Hennessy et D.A. Patterson, "*Computer Architecture, a Quantitative Approach*", Morgan Kaufmann Publishers, second edition, 1996
- [2] J.H. Collet, D. Litaize, J. Van Campenhout, C. Jesshope, M. Desmulliez, H. Thienpont, J. Goodman, and A. Louri ; "*Architectural approach to the role of Optics in monoprocessor and multiprocessor machines*", Applied Optics, **39**, 671-682, (2000)
- [3] M. Papamarcos, and J. Patel, "*A Low Overhead Coherence Solution for Multiprocessors with Private Cache Memories*", in proceedings of the 11th Annual International Symposium on Computer Architecture, pp. 348-354, June 1984; IEEE publish NY, ISSN 0149-7111
- [4] Jim Handy, "*the Cache Memory book*", Academic Press, second edition 1998, San Diego, ISBN: 0-12-322980-4
- [5] P. Sindhu, J.M. Frailong, J. Gastinel, M. Cekleov, L. Yuan, B. Gunning and Don Curry, "*XDBus: A High-Performance, consistent, Packet-Switched VLSI Bus*", in IEEE proceedings of Computer Conference (COMCON), Spring 93, pp 338-344; IEEE Society Press, Los Alamitos, ISBN: 0-8186-3400-6
- [6] M. Galles, and E. Williams, "*Performance Optimisations, Implementation and Verification of the SGI Challenge Multiprocessor*", IEEE Proceedings of 27th Annual Hawaii International Conference on Systems Sciences, pp 134-143, (1994); IEEE Society Press
- [7] Technical Report:www.sun.com/datacenter/products/e6000.htm
- [8] W. Hlayel, D. Litaize, L. Fesquet, and J.H. Collet., "*Optical versus electronic bus for address-transactions in future SMP architectures*", Proc. International Conference on Parallel Architectures and Compilation Techniques (PACT), Paris october 1998, p 22-29; Editor: IEEE Computer Society, ISBN: 0-8186-8591-3
- [9] S.C. Woo, M.Ohara, E. Torrie, J.P. Singh, and A. Gupta, "*The SPLASH-2 Programs : Characterization and Methodological Consideration*", Proc. International Conference on Computer Architecture (ISCA), Santa Margherita Ligure, Italy, 22-24 June 1995, pp.24-36.
- [10] Texas Instrument , "*Fast GTL Backplanes with the GTL1655*," Application report: SCBA015;january 1999
- [11] H. Johnson et M. Graham, "*High speed digital design*", Prentice Hall PRT, 1993, ISBN: 0-13-395724-1
- [12] D.A.B. Miller, "*Physical reasons for optical interconnection*", International Journal of Optoelectronics, Vol 11, No 3, p155-168, (1997)
- [13] G. Yayla, P. Marchand, and S. Esener, "*Energy and speed analysis of digital electrical and free-space optical interconnections*", in Optical Interconnections and Parallel Processing: The Interface: Edited by A. Ferreira and P. Berthome eds., Kluwer, chap. 3, (1997)
- [14] M. Kuijk, D. Coppee, and R. Vounckx; "*Spatially Modulated Light Detector in CMOS with Sense-Amplifier Receiver Operating at 180 Mb/s for Optical Data Link Applications and Parallel Optical Interconnects Between Chips*", IEEE Journal of Selected Topics in Quantum Electronics, Vol. 4, No 6, p 1040-1045, Nov-Dec 1998.
- [15] M.W. Haney, M.P. Christensen, P. Miljkovic, G.J. Fokken, M. Vickberg, B.K. Gilbert, J. Rieve, J. Ekman, P. Chandramani, and F. Kiamilev; "*Description and evaluation of the FAST-Net Smart Pixel-based Optical interconnection prototype*", Proc of the IEEE, vol 88, PP 819-828, (2000)

- [16] Y. Liu, E.M. Strzelecka, J. Nohava, M.K. Hibbs-Brenner, and Elia Towe; "*Smart Pixel Array Technology for Free Space Optical Interconnects*", Proc. Of the IEEE, Vol 88, pp764-768, (2000)
- [17] A. McCarthy, F. Tooley, E. Laprise, D. Plant, A. Kirk, M. Oren, and Y. Lu; "*Free-space optical interconnects system using polarization rotation modular arrays*", Proc. Optics in Computing, Quebec June 2200; SPIE vol 4089, p 272-277, (2000), SPIE: ISBN: 0-8194-3732-8
- [18] I. Redmond, and E. Schenfeld, "*Experimental results of a 64 channel, free space optical interconnection network for massively parallel processing*", Institute of Physics series number 139, proc. Optical Computing conference, pp 149-152 (1994)
- [19] H. S. Hinton, and T.H. Szymanski, "*Intelligent Optical Backplanes*," IEEE proceeding of the 2nd international conference on Massively Parallel Processing using Optical Interconnects (MPPOI'95); pp133-143, (1995); Editor: IEEE Computer Society Press ISBN: 0-8186-7101-7
- [20] E. Berglind, L. Thylén, B. Jaskorzynska, and C. Svensson, "*A comparison of Dissipated and Signal Power-to-Noise Ratios in Electrical and Optical Interconnects*"; J. of Lightwave technology, vol 17, pp 68-73, (1999)
- [21] C. Qiao, and R. Melhem, "*Time Division Optical Communications in Multiprocessor Arrays*", IEEE Trans. on Computers vol 42, 577-590 (1993)
- [22] A. Guo, R. Melhem, R. Hall, D. chiarulli, and S. Levitan, "*Array processors with pipelined Optical busses*", J. Parallel Distributed Computing vol 12, 269-282, (1991)