Performance Evaluation of the ATOLL Interconnect

Holger Fröning, Mondrian Nüssle, David Slogsnat, Patrick R. Haspel, Ulrich Brüning

Computer Architecture Group, University of Mannheim, Germany
D-68131 Mannheim, B6, 26-29
email: {holger,mondrian,david,patrick,ulrich}@ra.ti.uni-mannheim.de

Abstract

The ATOLL System Area Network (SAN) is a low latency interconnect that integrates all required components of a communication system for cluster computing into a single chip. The chip integrates not only four communication devices but also a high performance crossbar. The low latency and the scalability of the ATOLL chip implementation makes it an interesting technology in the area of cluster computing. The following paper presents performance results of basic point-to-point benchmarks together with an evaluation of the use as a cluster interconnect.

Keywords: Interconnection Networks, Cluster Computing, High Performance Computing, Performance Evaluation

1 Introduction

Cluster computing is becoming an increasingly important trend in high performance computing due to the steady cost-reduction and performance increase of the key computing resource used: the PC. Mass production makes the PC a cost efficient and therefore attractive building block for parallel computing systems. Computing can be performed at very high performance levels by these 'Gigahertz' processors, but the communication between the PCs, used as nodes, is mostly underdeveloped. Using the inexpensive Gigabit Ethernet or upcoming 10-Gigabit Ethernet for communication may satisfy some applications but many demand much higher communication bandwidth and, more importantly, much lower latency as the (10-)Gigabit Ethernet can provide [1]. This is the driving force behind the development of system area networks (SANs), which tries to trade low latency and high bandwidth versus interconnect length and costs. One successful representative of such a SAN interconnect is Myrinet 2000 [2]. Other players in the area of high-speed interconnects are Infiniband [3] and Quadrics Network QsNetII [4]. The QsNet with the ELAN4 version is one of the most sophisticated interconnect as it includes virtual memory support by a translation look aside buffer (TLB) and programmability by two on-chip processors, a thread CPU and a μCoded CPU. However, its high cost presents a challenge in today's commodity-driven market. In addition, unlike ATOLL it still requires external components to implement larger network installations.

The ATOLL research effort optimizes both components of communication performance, latency and bandwidth, and keeps the cost at the lowest possible level [5]. This is achieved by implementing all required components (host interface, host ports, network interfaces, switch, link interfaces) for a SAN on a single chip (except the link cables) combined with very innovative architectural features at a cost/performance ratio not yet achieved. The ATOLL chip with a die size of 5.7mm by 5.7mm was designed by the Computer Architecture Group at the University of Mannheim [6] and is produced by UMC using 0.18mm generic CMOS technology.

A software environment developed for the ATOLL chip, which contains the application programming interface PALMS, the MPI 2.0 port of MPICH [7][8], an ATOLL daemon for setup and control of the interconnect and an ATOLLrun access software, all available as open-source software [9].

In the following chapter we will describe the basic architecture of ATOLL including the aspects of scalability. In chapter 3 the test methods for the performance evaluation are introduced. Chapter 4 presents the test environment and the achieved performance results regarding bandwidth and latency for the ATOLL interconnect in detail. Chapter 6 concludes our description, summarizes the state of the project and provides an outlook.

2 ATOLL Basic Architecture

ATOLL is a complete 'network on a chip' designed for use in high-performance clusters. The requirements for low latency and high bandwidth are satisfied by using techniques like wormhole-switching, source-path-routing and user-level-communication [10]. ATOLL was specifically designed for use in high-performance SANs with fault-tolerance as well as guaranteed and in-order message delivery in mind. In addition to the requirements for high performance, cost reduction was a first-order concern [5].

ATOLL makes use of wormhole switching and source path routing. Wormhole switching enables the lowest latencies possible while also minimizing buffer size requirements. Source path routing is inevitable for low-latency communication, too. Software layers can precalculate routing strings to avoid look-up penalties at each switching element. Each hop consumes the first routing byte. Switching elements determine the output port by inspecting the first routing byte without table lookups, thus making hardware logic compact and very fast. The low fall-through latency achieved when traversing a node from link input to link output is a result of the consequent use of these techniques.

A complete ATOLL interconnect consists only of the network interface cards (NICs) and cabling. The chip on the NIC completely integrates all components required for integration towards host and network side [11]. Due to the integrated crossbar, no additional external switching hardware is required.

This distributed crossbar approach has many advantages. First, scaling the ATOLL network is extremely simple. Only NICs and cabling have to be added. The additional NICs add all required switching capability. Second, software can monitor and control the crossbar via control, status and debug logic. Thus the network traffic can be supervised directly and in simple and efficient way. Third, there is only one component required to build the network, lowering maintenance efforts. Basic communication functions are directly implemented in hardware.

As seen in figure 1, each crossbar provides 4 ports to the host side (host ports) and four link to the network side (link ports). Each host port can be mapped into user address space for direct user-level communication.

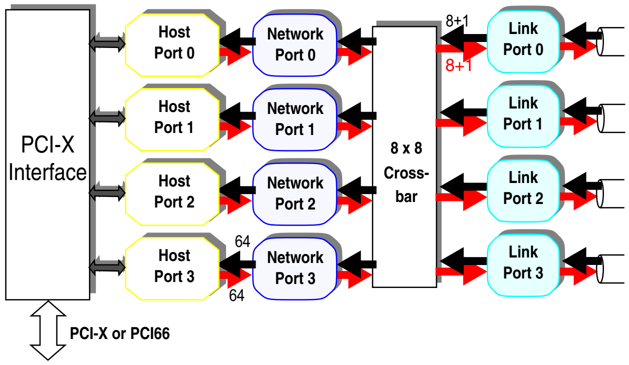

Figure 1 Top-level ATOLL chip structure

2.1 Topology

The preferable topology for the ATOLL network is a static one, because all switching capability is located inside the NICs. No external switching stages are required. The number of four links per NIC permits 2D topologies of grids or tori. Because of the shorter diameter of a torus in comparison to a grid [12], the 2D-torus is the recommended topology. An example 2D-grid with 16 nodes is shown in figure 2, which can easily be scaled to larger numbers of nodes.

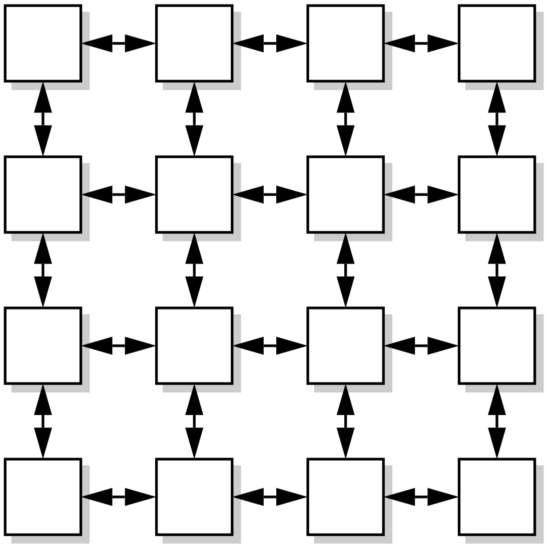

Figure 2 Example topology with 16 nodes

It should be mentioned that both regular and irregular topologies are supported from the hard- and software side. This is an important feature regarding fault-tolerance, since the failure of any element of the network turns a regular topology into an irregular one.

Topologies requiring several hops to reach a destination are very sensitive to fall-through latencies. In the ATOLL interconnect, the penalty for each hop is minimized. Within only 27 clock ticks a hop is traversed. This is as low as 90 nsec for each hop for a clock frequency of 300 MHz. Most of these 27 ticks are needed to re-synchronize the incoming data to the core clock.

According to [12], the number of cables for a 2D-tori is twice the node count. Compared to a central-switched network, which has a link count equal to the node count, it is higher but provides more switch over facilities. Considering a central switched network with two links from each node to the central switch for improved reliability, the link count is even equal.

For smaller systems, other topologies are even more adequate. For systems up to 5 nodes, a fully-interconnected network is possible, providing the minimal diameter of 1.

2.2 Scalability

The term scalability can refer to many aspects of a network. Here, scalability regarding the size of the network, addressing and performance is considered. These three aspects are now treated in more detail.

The major aspect regarding scalability in size is the required switching capability. Because every ATOLL card adds all the switching capability required for the integration into the network, the network is scalable without limitation. Regarding scalability of static networks, the node degree has to be invariant of the size of the network. The preferable topology for the ATOLL interconnect is a 2D-torus, and this topology has a fixed node degree of four. Thus, a typical ATOLL network with a 2D-torus topology has no limitations regarding scalability of network size.

For a source-path-routed network the major issue regarding scalability is the maximal length of the routing string. In fact, for the ATOLL network the length of the routing string is practically unlimited[1] and consequently provides adequate support for a scalable network.

Regarding the scalability of performance, the first issue is the end-to-end latency. Latency increases with each additional hop by 90 nsec, as mentioned before. For example, the network diameter of a 2D-torus with 256 nodes is 8 [12]. Thus, in worst case the start-up latency is increased by 7 hops or 0.63 µsec, which is negligible compared to other latencies in the communication path. The bandwidth is not affected by the number of hops due to the pipelined crossbar as switching element. Unlike interconnects using bus structures as building blocks, here the full bandwidth is available to the four bidirectional channels in and out of the NIC. The only issue affecting bandwidth is blocking of messages. Long messages can block a lot of intermediate crossbar stages. This problem can be diminished by reducing the maximum size of the packet length. Per-channel FIFO input buffers in front of the crossbar reduce the effect of blocking and provide a smooth transition from wormhole switching to store-and-forward in case of a heavy loaded interconnect.

3 Test Methods

3.1 Basic tests

First of all, the two test methods used here are introduced together with their intention regarding measurement results. These two methods are the ping-pong test and the streaming test.

In the ping-pong test, one node - the master node sends a message to another node - the slave node. When receiving the message, the slave node immediately returns it to the master. The master receives the returned message and measures the time passed from sending until reception of the message. Obviously, the time spend for the sending and reception on the master side is included in this time. Half of this time is commonly known as half round-trip latency. For minimal message size, this corresponds to the start-up latency. The resulting bandwidth of this kind of test is, due to interfering sending and receiving data streams, not very meaningful to evaluate the performance of a network.

The streaming test also involves two nodes. Here the master node sends messages to the slave node, which only consumes them without any response. The intention of this test is to measure the available effective bandwidth.

Using these two methods, the most relevant data regarding transmission characteristics can be determined: bandwidth and latency of the network. Generally, start-up latency and peak bandwidth are of most concern. Peak bandwidth is normally achieved with very large messages. The intense of this paper is to set another focus: the gradient of the bandwidth-message size graph. This shows in detail the relationship between message size and bandwidth.

3.2 Advanced tests

In comparison to the basic tests described above the advanced tests involve more than two nodes. The costs per hop, which is in fact a very interesting point especially in static interconnects like ATOLL, is not treated by the basic tests. The costs per hop introduced with this approach must be taken into account and require a closer analysis. Most networks for cluster use are not direct, i.e. one or more switching stages have to be traversed on the way from sending node to receiving node. To achieve scalability, most interconnects introduce additional switching stages out of separate building-blocks, e.g. crossbars.

Another important issue regarding performance, especially start-up latency, is the deviation of measured latencies. First, the average gives a good hint about the latency to be achieved. In addition, the variation of latency for subsequent messages provides more information. A plot is given showing the measured results for an amount of 10000 messages.

4 Performance Measurements

In the following, the measurement results are shown. The measurements using two nodes are accomplished on commonly available dual P4 XEON nodes. The CPUs run at 2.8 GHz, and use a Serverworks GC-LE chipset. The ATOLL cards are plugged into a dedicated PCI-X-100 slot. The nodes are equipped with 1 GB of RAM and run under Suse Linux version 9.1 using a standard kernel version 2.4.25. For tests including several nodes, those two systems are used as start and end node. Because of the ATOLL architecture, the internal architecture of the intermediate nodes is not relevant due to the host independent switching capability. However, for symmetry, these nodes are also

[1] In the descriptor of the message, the length of the routing string is limited to 2^{27} hops. The network layer does not limit the routing string.

running Suse Linux. They are equipped with dual P3 running at 1 GHz or dual P4 XEON running at 2.4GHz.

The core frequency of ATOLL is programmable up to 330 MHz. ATOLL was designed to run at 250 MHz, but recent tests show that nearly all chips are running with 300 MHz, thus in all tests ATOLL runs with 300 MHz.

4.1 Bandwidth measurements

As a first measurement, the bandwidth of ATOLL is presented. The bandwidth is measured using the streaming test. Messages sizes are varied from 1 byte up to 512 kbyte. However, only data up to 128 kbyte is presented since the measured bandwidth is constant for all larger message sizes. For each measurement point 1000 messages are transferred, the average bandwidth is plotted.

Figure 3 shows a peak bandwidth of 269 MByte/s. Note that the peak bandwidth is reached for a relatively small message size of 32 kbyte. For closer analysis, the detailed graph shows the bandwidth for small messages only. Half of peak bandwidth is exceeded with a message size of 512 bytes, while 4 kbyte messages achieve more than 90% of peak bandwidth. This shows in detail the performance for small messages.

Beside the streaming test over only one hop, the same test over several hops is measured. The previous analysis has shown that bandwidth is not be affected by the number of hops. In this experiment, the same streaming test is performed over 7 hops instead of only one hop.

The graph shows that indeed the bandwidth is independent of the number of hops. The slight increase of bandwidth can be attributed to minor deviations between the two measurements. The difference is only in the range of a few percents and likely results from caching or scheduling effects.

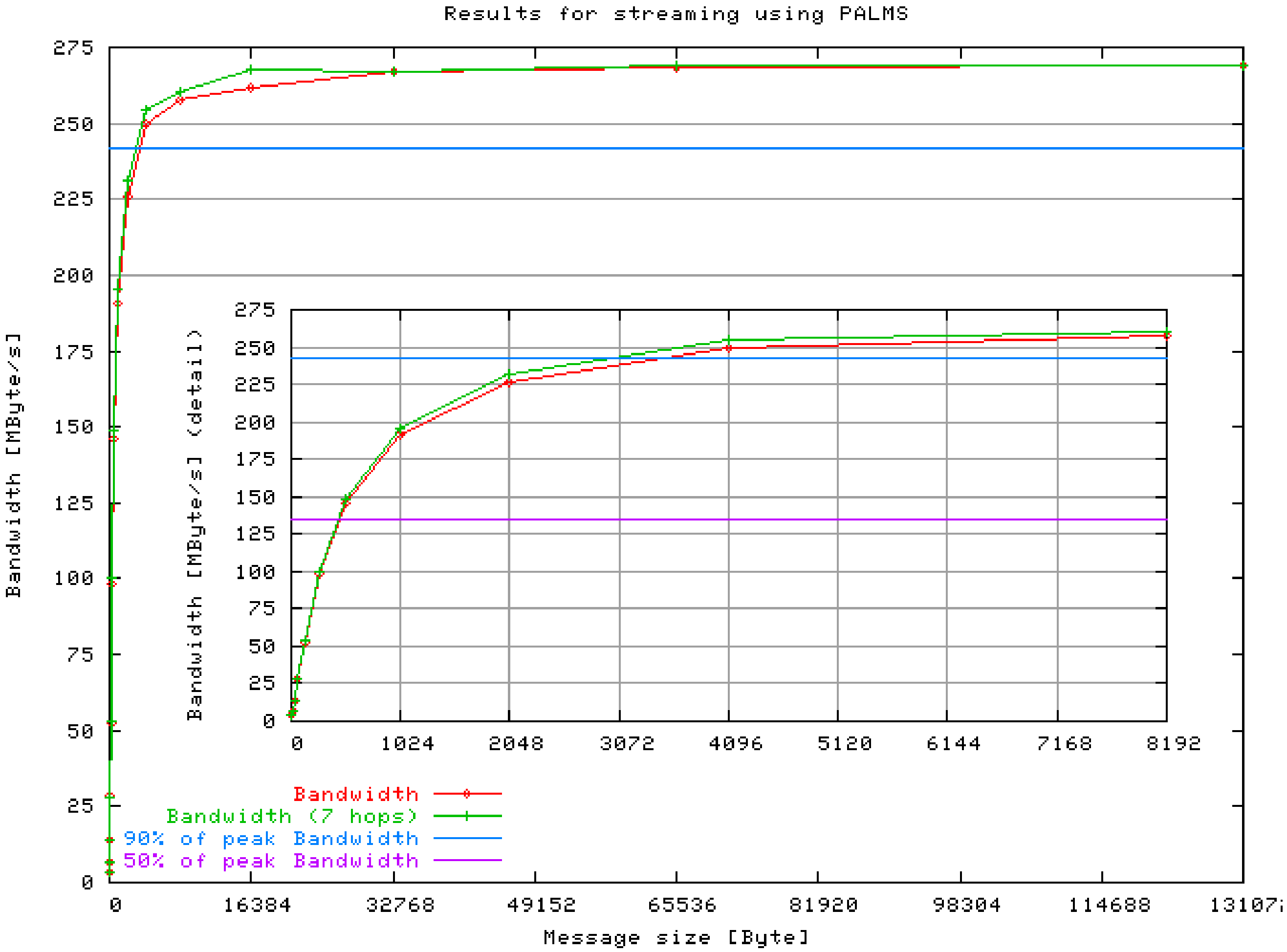

Figure 3 Bandwidth results for streaming

4.2 Latency measurements

Next, results from the ping-pong test over one hop are shown. Again, the message sizes varies from 1 byte to 512 kbyte, for each measurement point 1000 messages are transferred. Shown are the lowest, highest and average latency. The smaller graph shows the latency for small messages up to 128 byte in more detail.

The start-up latency is on average 3.4 µsec, which is a very good result compared to similar systems. In fact, the best case it is 3.34 µsec, only 0.06 µsec lower than the average. The small difference of only 2% shows that outliers occur very seldom and that the average latency is close to the minimum. The step to 4.1 µsec for a size of 8 byte is due to an additional memory access required at this message size. For the first message, the latency is very high. This can be explained with caching effect due to a first reference. For example, a message with the size of a cache line (64 byte) is still transferred with a latency below 5 µsec.

The number of hops must be taken into account when considering latency. The measurement above is the ping-pong test over only one hop, showing the lowest possible latency. For the next measurement, the ping-pong test is run over 7 hops. For comparison, the results of the previous test are plotted again. The expected increase is 27 clock ticks per hop, or 90 nsec at a core frequency of 300 MHz. Compared to the previous measurement, additional 6 hops are traversed, resulting in an expected latency increase of about 540 nsec.

According to the graph, the increase is nearly constant for all message sizes. For message sizes of 1 byte latency increases by about 560 nsec and for 8 byte by about 510 nsec. This is very close to the calculated value of 540 nsec and shows the linear latency increase.

4.3 Variation of latency

In the previous measurements, latency is shown using the best and worst time together with the average value. The small difference of average and best time is an indication that for most messages the latency is close to the best case. To show the latency variation in more detail, in Figure 5 the latencies of many consecutive messages is plotted. The message size is 8 byte, the measurement is performed over 7 hops.

The half round-trip latency is 4.15 µsec on average. The lowest latency measured here is 4.04 µsec, the highest 64.03 µsec. It can be seen that most message are close to the minimum latency, as expected. Only 3 of these 10000 measure point are over 10 µsec, only 15 are over 5 µsec. Thus, tolerating an increase of 10% of best latency (4.45 µsec), less than 2% (exactly 190 out of 10000) of the measurement points can be considered outliers.

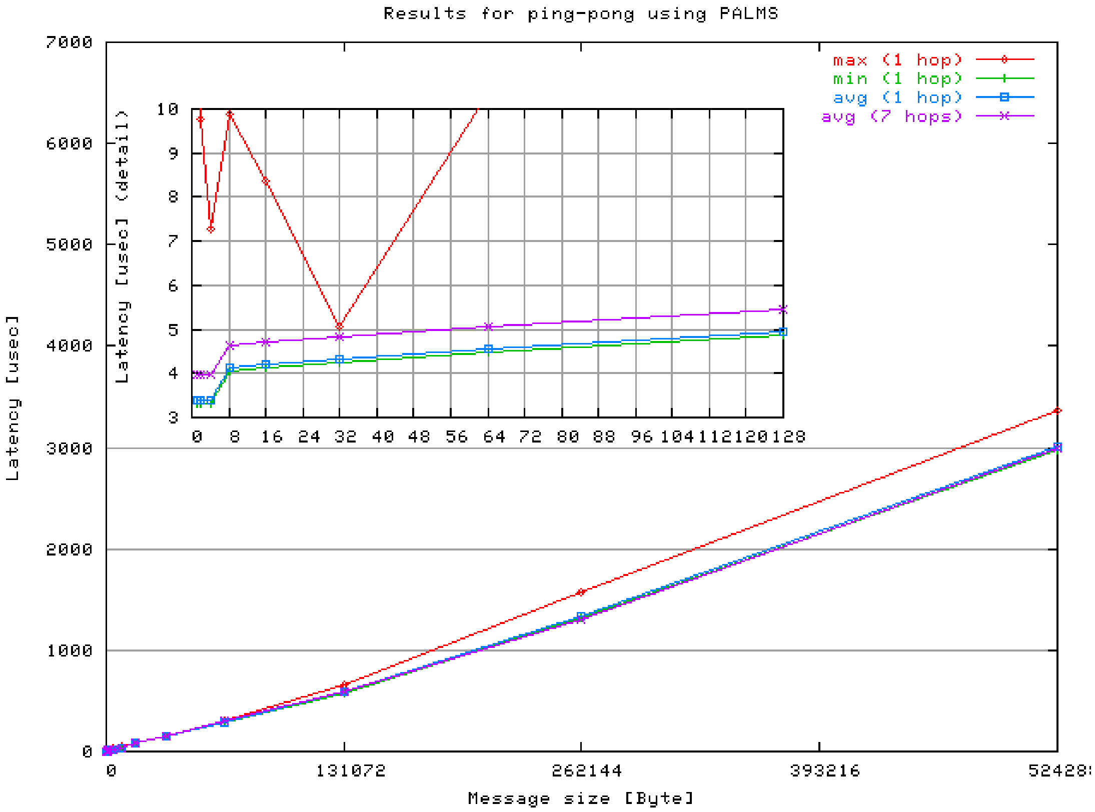

Figure 4 Bandwidth results for streaming

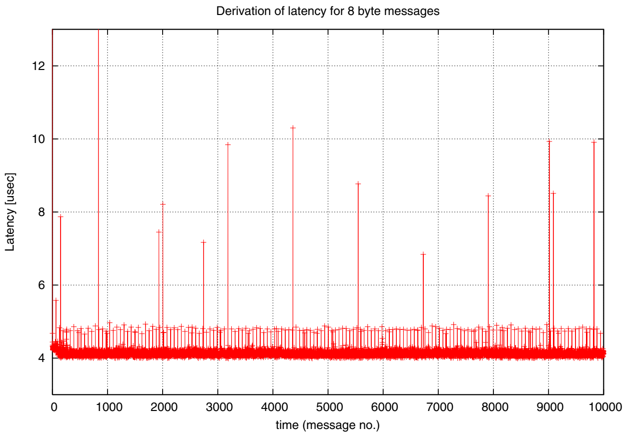

Figure 5 Latency variation for consecutive messages

5 Conclusion

Though the ATOLL architecture and chip implementation is a pure research project, the stated performance measurements prove ATOLL as a competitive interconnect solution for parallel computing. With a start-up latency of 3.4 µsec in average, ATOLL is a very successful research project. ATOLL reaches more than 90% of its peak bandwidth at a message size of 4 kbyte, which enables massive fine grain parallel computing only known from traditional super computers. Other communication schemes which include large bulk transfers, benefit more from the available bandwidth. ATOLL shows a peak bandwidth of 269 MByte/s over one link.

Furthermore, ATOLL's fall through latency of 90 nsec and the distributed switching resource ensures scalability in terms of network size and minimizes the latency to traverse the network. This has been achieved by a set of careful architectural design decisions, from wormhole switching, network packet design and source path routing to user-level communication, and by a fully pipelined and coding style aware hardware RTL design and implementation.

The most important part of ATOLL is the distributed crossbar, due to which no external and expensive switching elements are necessary.

The entire chip implementation was done by the Computer Architecture Group at the University of Mannheim and their research partners.

In the past, the ATOLL architecture has been presented in detail. This work shows the results of micro-benchmarks, which are limited to point-to-point connections. The next step is to benchmark a complete system using ATOLL as interconnection network. This will show the impact of the distributed crossbar as switching resource in more detail, providing benchmark results to compare the ATOLL performance to other widespread cluster interconnects.

6 Acknowledgements

We would like to express our appreciation for the close collaboration with the VCAD Services of Cadence Design Systems that enabled us to handle the physical design of ATOLL [13].

We also would like to thank the High Performance Technical Computing group of Hewlett Packard for providing us an Opteron-based cluster for future bench marking. Last but not least we would like to thank our graduate students for their excellent work.

7 References

[1] NetPIPE, http://www.scl.ameslab.gov/netpipe.

[2] Myricom: A Gigabit-per-Second Local Area Network, *IEEE Micro*, 1995.

[3] Voltaire, http://www.voltaire.com.

[4] QsNet: The Quadrics Network, Quadrics Supercomputer Ltd., Jan. 1999, http://www.quadrics.com

[5] Ulrich Brüning, Holger Fröning, Patrick R. Schulz, Lars Rzymianowicz, ATOLL: Performance and Cost Optimization of a SAN Interconnect, *IASTED Conference: Parallel and Distributed Computing and Systems (PDCS)*, Nov. 4 - 6, 2002, Cambridge, USA.

[6] Computer Architecture Group, University of Mannheim, http://ra.ti.uni-mannheim.de

[7] W. Gropp, E. Lusk, N. Doss, and A. Skjellum, A high performance, portable implementation of the MPI message passing interface standard, *Parallel Computing*, vol. 22, pp. 789--828, Sept. 1996.

[8] MPICH - A portable implementation of MPI, http://www-unix.mcs.anl.gov/mpi/mpich.

[9] ATOLL, http://www.atoll-net.de.

[10]L. Rzymianowicz, U. Brüning, J. Kluge, P. Schulz and M. Waack, ATOLL: A Network on a Chip, *Cluster Computing Technical Session (CC-TEA) of the PDPTA'99 conference*, June 28 - July 1, 1999, Las Vegas, NV. (3).

[11]Jörg Kluge, Ulrich Brüning, Markus Fischer, Lars Rzymianowicz, Patrick Schulz and Mathias Waack, The ATOLL approach for a fast and reliable System Area Network, *Third Intl. Workshop on Advanced Parallel Processing Technologies (APPT'99) conference*, Changsha, P.R. China, October 19-21 1999.

[12]Kai Hwang, Zhiwei Xu, *Scalable Parallel Computing*, p. 293, McGraw-Hill, 1998.

[13]Patrick R. Schulz, Ulrich Brüning, Gunter Strube, SEED2002: Support of Educational course for Electronic Design, *IEEE International Conference on Microelectronic Systems Education (MSE)*, June 1-2, 2003, Anaheim CA, USA.