2007－MPS－65 (11)
2007／6／25

メモリスロット直結型ネットワークインタフェースへのマルチキャストの実装と評価

太田 淳[†] 濱田芳博[†] 北村 聡[††]
田邊 昇[†††] 天野英晴[††] 中條拓伯[†]

現在，コンシューマ向けパーソナルコンピュータを多数利用したグリッドクラスタが普及している．しかし，ネットワークインタフェースの性能はプロセッサの性能向上に追いつきにくい現状にある．この背景から，我々はメモリスロットに装着するネットワークインタフェース DIMMnet を提案している．今回我々は，DIMMnet に対して InfiniBand のハードウェアマルチキャストを実装し，多数のクラスタに対して同一のデータを送出する場合の性能を検証する．

Implementation and Evaluation of Multicast Mechanism on Network Interface Plugged into a Memory Slot

Atushi Ohta,[†] Yoshihiro Hamada,[†] Akira Kitamura,[††] Noboru Tanabe,[†††] Hideharu Amano[††] and Hironori Nakajo[†]

Recently, clusters which use many personal computers are widely used. However, performance of network interface does not catch up with the performance improvement of processors. Considering the background, we have introduced network interface DIMMnet which is plugged into a memory slot. In this paper, we have implemented hardware multicast of InfiniBand for DIMMnet. This implementation would improve performance in the case when the identical data are sent out to multiple clusters at the same time.

1. Introduction

Fast network interface for clusters has emerged with the performance improvement of the processor so far. However, as for the communication transfer rate of the network interface, it is difficult to catch up with the improvement of the performance of recent processors.

In order to overcome the problem, we propose network interface DIMMnet which uses the memory slot interface whose delay is lower than conventional I/O buses such as PCI bus interface. Progress of the memory bus architecture for personal computers is remarkable. Moreover, a memory slot exists in a common computer, thus DIMMnet has an advantage to be equipped in most personal computers among nodes in a cluster.

There are communication libraries such as Message Passing Interface (MPI)[1] for parallel computing. An MPI library has these functions of the group communication. The group communication uses the iteration of the point-to-point communication or a hardware multicast. This performance of the group communication will be improved with the multicast mechanism of InfiniBand in DIMMnet.

In this paper, we have implemented multicast communication specified in specification of InfiniBand in a DIMMnet[2] network interface as the first step.

This mechanism is implemented by embedding function of hardware multicast in the network interface InfiniSWIF[3] within FPGA of DIMMnet-2[4]. And also the Subnet Management Agent software which handles a management packet. Then, we show multicast communication performance in DIMMnet-2, and comparison between throughput of multicast and unicast that transmits a message to the target node in turn with point-to-point communication.

2. Network Interface DIMMnet

Network interface DIMMnet which is plugged into a memory slot is being developed for three generations.

† 東京農工大学工学府
Tokyo University of Agriculture and Technology
†† 慶應義塾大学理工学部
Faculty of Science and Technology Keio University
††† 株式会社 東芝 研究開発センター
Corprate Research and Development Center, Toshiba

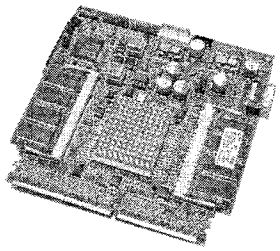

図 1 DIMMnet-2 試作基板
Fig. 1 DIMMnet-2 prototype board

The lastest generation, DIMMnet-3 is still under development. Therefore, implementation in this paper is targeted to DIMMnet-2 of the second generation in the DIMMnet series.

Development of DIMMnet-2 started under collaborator of Tokyo University of Agriculture and Technology, Keio University, Wakayama University and Toshiba Corp. in 2002. Figure 1 shows the prototype board of DIMMnet-2.

DIMMnet-2 has the following features.

- Plugged into DDR SDRAM memory slot
- InfiniBand 4× connection
- Buffer memory up to 8GB
- Pre-fetch mechanism

Configuration in FPGA Virtex-II Pro XC2VP70 which is mounted on the DIMMnet-2 board consists of four modules.

- Core Logic
  The main function of DIMMnet-2. When it receives commands from a processor, reads and writes data in buffer and perform pre-fetch function.
- DDR SDRAM Interface
  Controls accessing between the motherboard and the Core Logic.
- InfiniSWIF
  A interface between Core Logic and the InfiniBand network.

3. Overview of InfiniBand

InfiniBand is a network architecture which is specified on by InfiniBand Trade Association (IBTA)[5] in 2000. The specifications of InfiniBand are defined by InfiniBand Architecture Specification, and current version is Release 1.2[6] announced in 2004.

InfiniBand has sixteen logical links called "Virtual Lane" within a physical link. They have priority, and then Quality of Service (QoS) is achieved by using differences in its priority.

A network management packet controls local subnet. The following agents, which are executed in upper layer than transport layer, manage network.

- Subnet Management Agent
  Subnet Management Agent (SMA) exists in every device which is equipped in whole InfiniBand subnet.
- Subnet Manager
  One or more Subnet Manager (SM) must exist in the subnet. It assigns the Local ID (LID) to identify each device in the subnet.

SMA and SM sends and receives packets via VL15 in a Virtual Lane.

3.1 Hardware Multicast in InfiniBand

When a packet including Destination LID (DLID) assigned to multicast is sent to an InfiniBand switch, the switch recognizes the packet as a multicast packet.

The recognized multicast packet is duplicated in the switch and transferred to a specified port. The port for multicast is specified by a multicast table in the InfiniBand switch.

A multicast table whose size is 512 Bytes in an InfiniBand switch is set by network management packet from an SMA.

4. Design of the Multicast Mechanism

It is necessary to set up DLID of a multicast packet with value for multicasting and send it as well as to rewrite a multicast table of an InfiniBand switch through a management packet for hardware multicast of InfiniBand.

4.1 Identification of a Multicast Packet

In order to enable sending of a multicast packet, we have coded a function which overwrites original DLID to multicast DLID into network interface InfiniSWIF of DIMMnet-2.

A packet format is defined for Core Logic and InfiniSWIF. Header of a packet has a flag, called as MCP flag, which indicates that the packet is for multicasting for the future extension.

Without using either DLID of InfiniSWIF for multicasting or an MCP flag, a multicast communication is possible by setting DLID for multicasting when Core Logic sends the packet to InfiniSWIF. However, considering changing specification of DIMMnet in the future, MCP flag is adopted for a packet to be identified as a multicast packet.

4.2 Update Multicast Table of a Switch

A function that updates a multicast table is realized by adding some code to SMA in an FPGA of DIMMnet-2.

The SMA is executed on PowerPC processor inside the FPGA whose peripheral modules are wrapped by Verilog modules which are developed with Xilinx Embedded Development Kit (EDK)[7] in the FPGA are wrapped Verilog modules. A function of sending and receiving management packets has already been equipped in an SMA because it is needed to send and receive management packets for initialization such as getting Local ID from an SM. These management packets are used to set a multicast table.

5. Performance Evaluation

The result of performance evaluation of the InfiniBand multicast mechanism which is coupled into InfiniSWIF is shown in this section. The effect of the multicast mechanism is shown by comparing performance with communication of the unicast communication which transfers a message to other node in turn, and ideal point-to-point communication performance. This performance evaluation shows a bandwidth and latency at their ways.

5.1 Evaluation Environment

Our experimental testbed cluster connects for PCs, and each PC equips with DIMMnet-2 via an InfiniBand switch. The InfiniBand switch is ISR-6000 SW-IB4 of Voltaire, Inc[8]. To evaluate performance, internal logic in an FPGA of DIMMnet-2 consists of a test module which conducts a packet, InfiniSWIF and an SMA program.

One PC of a four-node cluster is used to transfer a test packet. For point-to-point communication, the time since a packet is sent from a sending node till the packet is received by a receiving node is measured. For unicast communication, the time since a sending node sends a packet to other three nodes in turn till the third node receives the packet is measured. The time period of multicast communication measures from sending the multicast packet to receiving the packet for a 3 nodes cluster.

Sending data size in evaluating throughput varies from 8 Bytes to 16 MBytes. Data under 2 KBytes is sent with changing the size of the packet. Data over 2 Kbytes is sent with continuous packets each sized 2 KBytes. We have calculated throughput from the spent time and the data size for sending in each way of communication.

An evaluation method of latency is shown as follows. Latency of point-to-point communication is measured by the time since a send request of InfiniSWIF in a sending node is issued till a receive request of InfiniSWIF in a receiving node is issued. Latency of unicast communication is not measured since it is repetition of point-to-point communication. Latency of multicast communication is measured by the time since a send request of InfiniSWIF in a sending node is issued till a receive request of InfiniSWIF in the latest node in 3 nodes in a cluster.

5.2 Performance Evaluation Results

The measurement of throughput is shown in Figure 2. Multicast communication performs its throughput of up to 360MB/sec. Unicast communication performs around 260MB/sec and point-to-point communication performs its throughput of 780MB/sec. In the case of small data size, multicast communication performs its ideal throughput which is almost same as the throughput of point-to-point communication. However, the larger the data size gets, the lower the performance degrades. Degradation of throughput in multicast communication is large when the data size is 4KB that a packet is sent repeatedly. When data sized 16MB is sent, consumed time of multicast communication is twice as long as that of point-to-point communication. However, multicast communication performs high throughput in comparison with the unicast communication to three nodes.

In unicast communication, since packets are sent to nodes in turn, completion of the packet sending should be confirmed by polling processing. Expected performance of the unicast communication in small size packet is degrading under the influence of polling processing at a sending packet test code in SMA. However, the polling processing can ignored when transmitting a large packet.

Figure 3 shows latency of point-to-point and multicast communication. As shown in the figure, latency in multicast communication is 1.2–1.4 times as large as that of point-to-point communication. Latency gets longer in proportion with the size of a packet.

6. Study from the Results

Multicast communication increases processing of an InfiniBand switch. Therefore, increasing latency cannot be avoided. However, as for InfiniSWIF, there is little performance degradation which occurs in sending and receiving a multicast packet, because a node which sends multicast packet only sets up the destination LID by multicast LID. Furthermore, a receiving node can receive a multicast packet as a regular packet.

When the size of a multicast packet to be sent is small, latency by increasing copy processing of an InfiniBand switch is reduced due to a FIFO buffer in InfiniSWIF and a latency in the transmission. However, performance degrades drastically when multicast packets are transmitted repeatedly. The reason is that concealed latency appears in the interval of packet sending. Still the degradation of the perfomance in multicast communication is stemmed in sending a packet sized over 64KB.

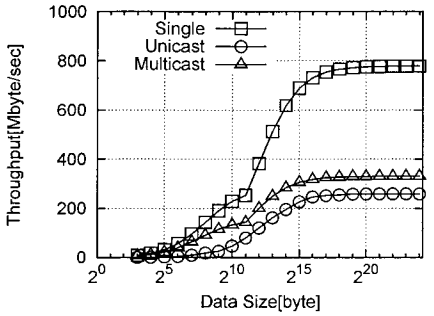

図 2 各通信方法におけるスループット
Fig. 2 Throughput of three transmitting ways

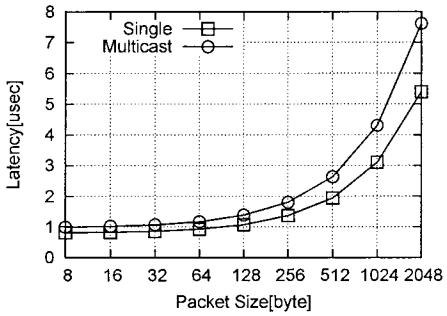

図 3 遅延の比較
Fig. 3 Comparing of Latency

7. Conclusion and Future Work

In this paper, we have implemented hardware multicast in the specifications of InfiniBand to the network interface DIMMnet-2 which plugged into a memory slot.

Performance of the new multicast communication, point-to-point communication and also unicast communication are compared.

Latency of the multicast communication increased slightly. When packet size is small, throughput of multicast communication increases approximately 1.4–1.8 times in comparison with point-to-point communication. When repeated packets are sent for huge data, throughput of multicast communication is degraded 2.5 times as low as throughput of point-to-point communication. However, the throughput is larger than unicast communication for three nodes.

We should improve two problems in our multicast implementation for DIMMnet-2.

First is to develop a device driver which software can use the InfiniBand hardware multicast of DIMMnet-2 in ease in order for software to use InfiniBand multicast communication on DIMMnet-2.

The second subject is to implement hardware reliability of InfiniBand on DIMMnet-2. The multicast communication of InfiniBand uses Unreliable Datagram (UD) communication in which an arrival of packets is not guaranteed from a limitation according to the specifications of InfiniBand. To overcome this problem, a mechanism of confirmation and retransmission of incoming packets have to be implemented into DIMMnet-2. The specification of InfiniBand defines the method of communications called Reliable Datagram (RD) in which an arrival of a packet is guaranteed. We will implement RD communication into DIMMnet-2 in order to send acknowledge for multicast packets.

The implementation will enable efficient multicast communication.

Acknowledgments This study was partly supported by MEXT Fund for Promoting Research on Symbiotic Information Technology.

References

1) Message Passing Interface (MPI) Forum Home Page : http://www.mpi-forum.org/
2) Noboru, T., et al. : Low latency communication on DIMMnet-1 network interface plugged into a DIMM slot, Parallel Computing in Electrical Engineering 2002, pp. 9-14 (2002).
3) Yoshihiro, H., et al. : A Packet Forwarding Layer for DIMMnet and its Hardware Implementation, Proceedings of The 2005 International Conference on Parallel and Distributed Processing Techniques and Applications(PDPTA '05), Vol. II, pp. 461-427(2005).
4) Noboru, T., et al. : Concept of DIMMnet-2 Network Interface Plugged into Memory Slot, in Japanese, IPSJ ARC-152, pp.61-66(2003).
5) InfiniBand Trade Association : http://www.infinibandta.org/home
6) InfiniBand Architecture Specification Release 1.2 : InfiniBand Trade Association(2004)
7) Platform Studio and the EDK : http://www.xilinx.com/ise/embedded_design_prod/platform_studio.htm
8) Voltaire, Inc. : http://www.voltaire.com/
9) Tanabe, N., et al. : Hardware Support for MPI in DIMMnet-2 Network Interface, Innovative Architecture for Future Generation High Performance Processors and Systems(IWIA '06), pp. 73-82(2006).