A Network Storage LSI Suitable for Home Network

Han-Kyu Lim*, Ji-Ho Han, and Deog-Kyoon Jeong*****

Abstract—**Storage over Ethernet (SoE) is a network storage architecture that allows direct attachment of existing ATA/ATAPI devices to Ethernet without a separate server. Unlike SAN, no server computer intervenes between the storage and the client hosts. We propose a SoE disk controller (SoEDC) amenable to low-cost, single-chip implementation that processes a simplified L3/L4 protocol and converts commands between Ethernet and ATA/ATAPI, while the rest of the complex tasks are performed by the remote hosts. Thanks to simple architecture and protocol, the SoEDC implemented on a single 4mmx4mm chip in 0.18um CMOS technology achieves maximum throughput of 55MB/s on Gigabit Ethernet, which is comparable to that of a high-performance disk storage locally attached to a host computer.**

Index Terms—**Ethernet, network, network processor, storage**

I. Introduction

Today, digital multimedia devices such as set-top boxes, DVD players, and HD-TVs, which require high volume storage, and home network demand to converge the storage and the network interface at home. True advantages of a home network are realized when storage devices dedicated solely for one application are offered for sharing among other systems for many different applications through the network. There, the network interface to such storage and application devices needs to be adequately designed for efficiently utilizing multimedia devices at home.

Consumer electronics devices are sold at a very low price. Furthermore, with rapid technology advancements combined with rapid growth of available contents, such devices are frequently supplemented with even cheaper devices with higher performance. Therefore, low cost is a primary requirement for home network devices while they should offer enough performance to support multimedia applications requiring much bandwidth. In our opinion, the performance of the network storage must be limited only by the performance of storage itself. That is, the network must perform like an IO and the storage should be able to yield performance comparable to that of a locally connected storage.

Although there have been many network storage devices, they are not well suited to application to home networks. A network file server is too expensive for home use. Although in some research [1] a network file server has been designed using embedded processor for purpose of reducing implementation cost, they did not offer high enough performance to support multimedia applications because of the bottleneck imposed by the software processing of the critical protocol. Universal Serial Bus (USB) mass storage [8] offered at a low cost for end users is a candidate for home network storage. However, it is very hard for the USB mass storage to be shared among multiple devices because of the inherent limitation that only one host must exist in every USB network.

In order to satisfy the requirements of the home network storage, we designed Storage over Ethernet (SoE), a network storage architecture. It offers convenience and efficiency like a network file server but it is designed to minimize the implementation cost. As its name shows, the SoE allows multiple devices to access the storage through the network without any server intervention.

Manuscript received November 23, 2004; revised December 8, 2004.
School of Electrical Engineering and Computer Science Seoul National University, Seoul, Korea
E-mail: *limbear@isdl.snu.ac.kr, **jhhan@isdl.snu.ac.kr
***dkjeong@isdl.snu.ac.kr

In this work, we describe the SoE Disk Controller (SoEDC) enabling commodity AT Attachment with Packet Interface (ATA/ATAPI) devices to be incorporated in the SoE architecture. Its architecture and implementation are optimized for low cost and high performance. As the SoEDC LSI chip integrated all components, except PHY, in only 16mm^2 die in 0.18 um CMOS technology, the SoE storage can be realized with only two chips, an Ethernet PHY chip and the SoEDC LSI chip. The SoE storage has been successfully demonstrated with the use of the SoEDC chip and shows the performance of 55MB/s, comparable to the disk locally and directly attached to a computer.

The organization of the paper is follows. The SoE system architecture will be described in Section 2 and the design of the SoEDC LSI will be covered in Section 3. The implementation results and measured performance of the SoEDC LSI will be followed in Section 4 and, finally, the paper will be concluded and summarized in Section 5.

II. SOE SYSTEM ARCHITECTURE

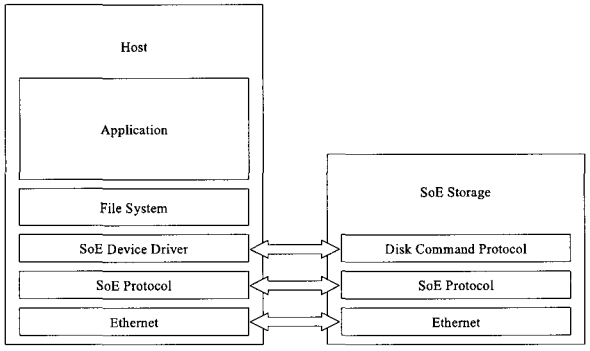


Fig. 1. SoE system architecture

Figure 1 shows the SoE system architecture. In this architecture, the SoE storage composed of an SoEDC and a storage plays the role of the server in the traditional network storage architecture [6], but it does not execute any file system on itself. To be more specific, SoE directly operates on a block, a raw material from which files are formed, like Storage Area Network (SAN) [7] architecture. Because the file system is offloaded from the SoE storage and moved to the remote hosts, tasks of the SoEDC are simple. This means that the SoEDC can be implemented in a simple hardwired control or in software on a low performance processor in achieving high performance.

Although the file system offload might make SoE look like a derivative of SAN, SoE has an apparent advantage over SAN. While SAN itself cannot be directly shared by multiple hosts, SoEDC can. In the SAN architecture, the applications on remote hosts can access the SAN only through the file servers directly attached to the SAN. However, in the SoE architecture, the applications can directly access the SoE storage through no server intervention. This is the reason why we named this architecture as SoE.

The operation of SoE is described below. Once an application on the remote host requests a file access through a system call, the file request are passed to the file system running on the same host. The file system fragments the file request into many primitive disk I/O requests and send its requests to a SoE device driver. The SoE device driver delivers the disk I/O requests to the SoEDC through an SoE protocol. The SoEDC operates in the same manner as the disk controller attached to the system except that it has an interface to the network.

III. DESIGN OF SOEDC

1. SoEDC LSI block diagram

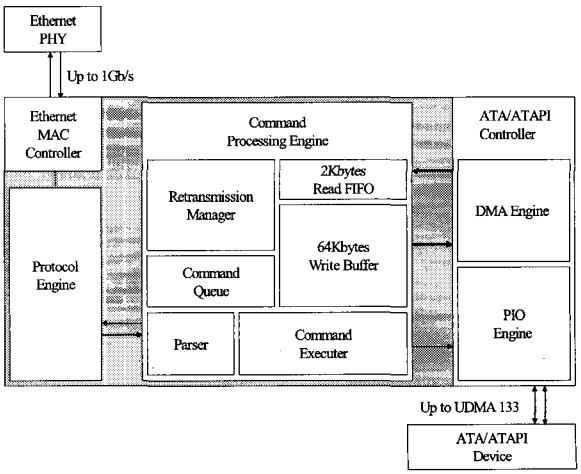


Fig. 2. Block Diagram of SoEDC LSI

The architecture of the SoEDC is shown in Figure 2 and consists of a gigabit Ethernet controller, a protocol engine, a command processing engine, a 2KB read FIFO, a 64KB write buffer, and an ATA/ATAPI controller. The protocol engine manages connections and information of remote hosts and performs en-capsulation and de-capsulation of protocol header. Depending on a requested command, the command processing engine manages the read FIFO and the write buffer which store data read from disk or written to disk. The ATA/ATAPI controller supports up to a 133MB/s bandwidth to cope with 1Gb/s network bandwidth.

At first, a host requests connection establishment and the protocol engine grants the request. After establishment, the host requests a disk I/O command through a command packet. After header of the received packet is de-capsulated in the protocol engine, payload of received packet is passed to the command processing engine as command. The command processing engine executes the I/O command controlling the ATA/ATAPI controller. When the operation is completed, the command processing engine sends a reply packet through the protocol engine.

2. Protocol Engine

As shown in Fig. 1, the SoE architecture does not use TCP/IP, the most commonly-used protocol on the network, but instead uses newly defined SoE protocol, we called LeanTCP. Although using TCP/IP stack has many advantages, it is not feasible for SoEDC because a hardware implementation of TCP/IP stack is impractical and a software implementation [4] requires a high performance processor in order to process packets at gigabit line rate. Some research [5] tried to implement TCP/IP in hardware, but they required much hardware because they focused on the applications of servers or routers where implementation cost is not important.

By simplifying TCP/IP, we defined LeanTCP protocol which provides reliable services like TCP/IP, but is much lighter than TCP/IP; LeanTCP name came from this. To lighten TCP/IP, some complicate but useless features such as frame reordering, flow control, and IP address are excluded from TCP/IP; frame reordering and flow control of TCP operate for routers existed in WAN and instead of IP address, MAC address is used to deliver packets in level 2 switching of LAN. Figure 3 shows leanTCP protocol header.

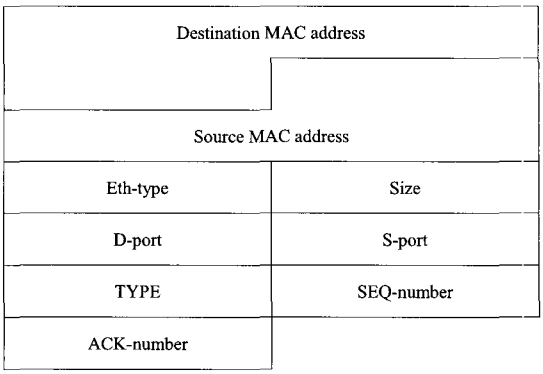


Fig. 3. Protocol Header

The Protocol engine is the hardware implementation of LeanTCP stack. It is charge of performing connection establishment and tear down, checking validity of incoming packets, retransmitting lost packets, and managing the information of remote hosts.

3. Command Processing Engine

The command processing engine composed of a command queue, a write buffer, a read FIFO, a parser, a command executer, and a retransmission manager executes disk I/O commands requested by remote hosts.

Before processing payload, the command processing engine fetches state information corresponding to an identifier, which is passed from the protocol engine from a state memory. The state information represents operation modes: the command and data modes. Initially, the command processing engine is in the command mode. In this mode, the command processing engine deals all received payloads as commands so that it sends all payloads to the parser for analysis. After parsing process, the command is queued in the command queue and waits until the command executer is idle. Receipt of a write command changes state from the command mode to the data mode. In the data mode, all subsequent payloads are saved in the write buffer and after completion, the state returns to the command mode.

IV. IMPLEMETATION AND PERFORMANCE

In order to benchmark the SoEDC performance, we experimented by using FPGA prototype board because the LSI is under fabrication. For comparison, the performance of three other devices, a local disk controller, an USB2.0 bulk storage controller [3], and 1Gb/s NAS [2] implemented in embedded processor, were also included.

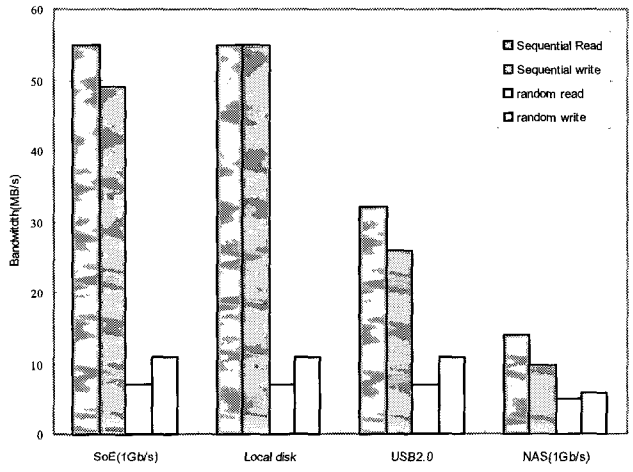


Fig. 4. Bandwidth

Table 1. Average Access Time

	SoE (1Gb/s)	Local Disk	USB2.0	NAS (1Gb/s)
Average Access Time (ms)	7	7	8	9

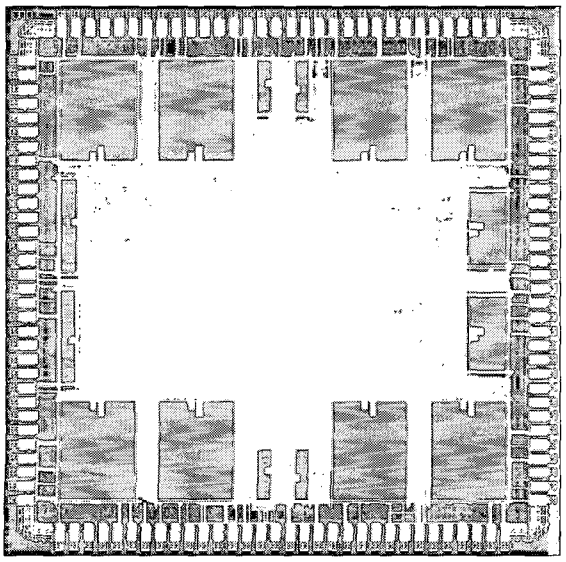


Fig. 5. Chip Layout

Figure 4 and table 1 show the performances of SoEDC. The measured performance of SoEDC was competitive to that of Local Disk Controller (LDC); The sequential read bandwidth is 55MB/s, the sequential write bandwidth is 49MB/s, the random read bandwidth is 7MB/s, the random write bandwidth is 11MB/s, and the Average access time is 7 ms. Each performance was equal to those of LDC except the sequential write bandwidth, which was decreased by a 10.9 % under LDC. However, we concluded that this degradation resulted from NIC because in the test using a server NIC, the sequential write bandwidth exhibited the same performance as LDC; generally, the network devices for desktops may not be optimized for up-loading affecting the write performance. Because SoE was targeted for home network, the experiment results using a server NIC were not included here. Although we considered the case of the desktop NIC, SoEDC overwhelmed other devices. Especially, even if the NAS had the network interface of 1Gb/s, it exhibited very low performance, 25.6 % of the LDC, because the used processor was the bottleneck in the NAS.

The SoEDC LSI is fabricated with 0.18-um six-layer metal CMOS process and housed in a 128-pin plastic QFP. The clock speed is 125 MHz. The chip layout of the SoEDC LSI is shown in Figure 5.

V. CONCLUSIONS

In this paper, a low cost and high-performance SoEDC LSI for home network applications is presented. This chip adopts a gigabit Ethernet MAC controller and an ATA/ATAPI controller supporting 133MB/s in order to achieve enough performance to support multi-media data. It is implemented in full hardware in order to lower power consumption, reduce die area, and achieve high-performance. The LSI performs the maximum performance of 55MB/s equal to that of local disk controller. In spite of full hardware implementation, it offers flexibility; the LSI supports all kind of ATA/ATAPI devices by only modifying the device driver. This chip is fabricated in a 0.18 um CMOS technology and the die area is 16 mm^2 including I/O cells.

REFERENCES

[1] A. Tomita, N. Watanabe, Y. Takamoto, S. Inohara, F. Maciel, H. Odawara, M. Sugie, "A Scalable, Cost-Effective, and Flexible Disk System Using High-Performance Embedded-Processors", Proceedings of the 2000 International Conference on Parallel Processing, pp 317-326.

[2] Buffalo, "Linkstation", http://buffalo.melcoinc.co.jp.

[3] Cypress Semiconductor Corporation, "EZ-USB AT2™ USB 2.0 to ATA/ATAPI Bridge", http://www.cypress.com.

[4] Hoskote, Y., et al, "A TCP offload accelerator for 10 Gb/s Ethernet in 90-nm CMOS", IEEE Journal of Solid-State Circuits, Nov. 2003.

[5] Schuehler, D.V., Lockwood, J.W., "TCP Splitter: a TCP/IP flow monitor in reconfigurable hardware", IEEE Micro, Jan.-Feb. 2003.

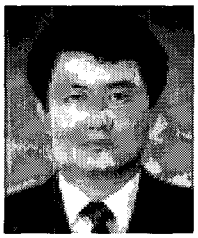
[6] S. Shepler., et al. "NFS version 4 Protocol", RFC 3010, Dec. 2000.

[7] Tom Clark, "Designing Storage Area Network", Addison-Wesley, 1999.

[8] USB Implementers Forum, Inc., "Universal Serial Bus Mass Storage Class Bulk-Only Transport", http:// www.usb.org.

Han-Kyu Lim received the B.S. and M.S. degrees in electrical engineering and computer science from Seoul National University, Seoul, Korea, in 2000 and 2002, respectively, where he is currently working toward the Ph.D. degree. His research interests are in the area of storage systems, network systems, and VLSI systems.

Ji-Ho Han received the B.S. and M.S. degrees in electrical engineering and computer science from Seoul National University, Seoul, Korea, in 2002 and 2004, respectively, where he is currently working toward the Ph.D. degree. His research interests are in the area of storage systems, network systems, and VLSI systems.



Deog-Kyoon Jeong is currently a Professor at the School of Electrical Engineering, Seoul National University, Seoul, Korea. He was the Director of Embedded Systems Research Center at the Seoul National University until the end of 2003. His main research interests include high-speed I/O circuits, VLSI systems design, high-performance network systems, and high-frequency RF circuits. He has B.S. and M.S. degrees in electronics engineering from the Seoul National University, and Ph.D. degree in electrical engineering and computer sciences from the University of California at Berkeley.