

Design of Multicore-based High Performance Load Distribution System

Joon Yeol Yoon*, Won Seok Choi**, Jong Oh Kim**, Jae Woo Ahn*, Jong Beom Kim*, Seong Gon Choi*

*Information & Communication Engineering, ChungBuk National University, Cheongju-si Chungcheongbuk-do, Korea

**Fisys Inc., 117 Dunsan-Daero Seo-gu Daejeon, Korea

maboloo@cbnu.ac.kr, wschoi@fisys.co.kr, jokim@fisys.co.kr, jwahni@cbnu.ac.kr, dragonslash@cbnu.ac.kr, choisg@cbnu.ac.kr

Abstract— This paper proposes a method of implementing a NIC (Network Interface Card) capable of data processing at 40Gbps. Recently, technologies such as cloud services, mobile, and big data have been developed and more and more traffic is increasing. Therefore, network equipment capable of processing such traffic at high speed is attracting attention. We developed a network card based on the TILE-Gx36 processor, which made it possible to process data onto high speed. As a result of our experiments, we confirmed that data transmission and reception are working normally at 40Gbps.

Keywords—40Gbps, TILE-Gx, by-pass, implementation, SDNIC

I. INTRODUCTION

Internet, mobile, big data and cloud-based services are growing more and more recently. As a result, traffic is rapidly increasing. The development of ICT (Information and Communication Technology) has also linked people, things, and services. And even the object-object connections will be possible in the future. Therefore, a huge amount of information travels through the network [1].

Recently, network equipment is being developed to solve such traffic problem. And there are various studies for efficient use of network equipment. One of them is SwitchBlade, a platform that integrates general packet processing functions of hardware modules and quickly deploys custom protocols to programmable hardware [2]. There is also a study on the mechanism of MLD (Multi-Lane Distribution) mechanism, which is a key implementation technology of high-speed ethernet [3]. There is another way to solve the delay problem that can occur due to the increase in traffic. There is also research in the SDN (Software Defined Network) controller to determine the QoS (Quality of Service) levels according to the type of application and to reduce the delay with the queue scheduling technique in the SDN switch [4].

In this paper, we propose a network interface card capable of high-speed data transmission. This is because the performance improvement in the hardware itself has a great effect on the performance of the entire system. The proposed network interface card provides various functions of packet processing and can process data onto a total speed of 40Gbps.

In addition, the bypass module can prevent data loss when the network card does not operate normally.

This paper is organized as follows. Section 2 describes the hardware configuration. Section 3 describes the operation method of high-speed data transmission. Section 4 describes the testing environment and results. Section 5 is the conclusion.

II. HARDWARE CONFIGURATION

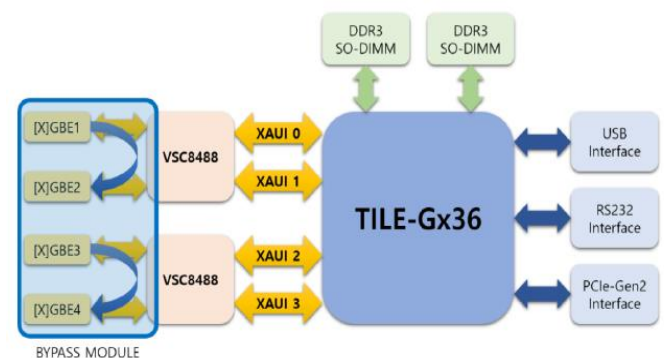


Figure 1. Proposed Network Interface Card Architecture

Figure 1 shows a block diagram of the proposed architecture of a TILE-Gx network interface card. The TILE-Gx processor consists of 36 cores of RISC processors. And optimized for networking and multimedia applications. It is also ideal for sophisticated networking applications requiring 20 to 40Gbps [5]. TILE-Gx has two 8GB memory DDR3 SO-DIMMs connected. And TILE-Gx has four 10Gbps XAUI ports. TILE-Gx and VSC8488 are connected through the XAUI port. VSC8488 provides the highest port density of the lowest cost resulting in 40Gbps performance. In addition, a bypass module can be added to prevent data loss when the card does not operate normally.

The TILE-Gx processor consists of 36 cores of 1.2 GHz, mPIPE (Multi-process Pipeline) with packet classification, RAM, encryption acceleration engine MiCA for encryption, and TRIO (Transactional I/O) to control DMA (Direct Memory Access) data movement and buffer management between PCIe (Peripheral Component Interconnect Express).

DDR3 SO-DIMMs are added to support ECC (Error-Correcting Code) Memory function, which detects and corrects internal data corruption better. Also, adding bypass module to prevent data loss in the entire system when a system problem occurs. Use a heatsink and fan for the cooling function and finally install the ATX 12V Power Connector for power supply purpose. Network connections use SFP+ (small form-factor pluggable). SFP+ is an optical module transceiver supporting 10Gbps.

III. METHOD FOR HIGH-SPEED DATA TRANSMISSION

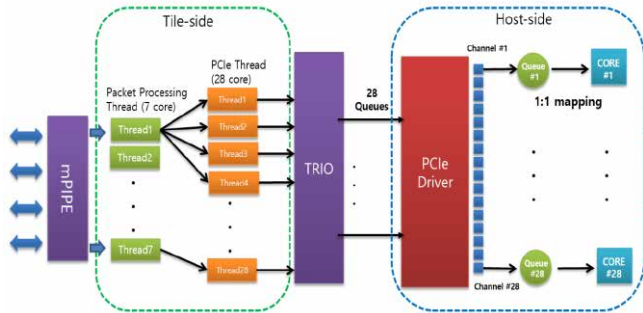


Figure 2. Packet processing architecture for 28-cores

A. Components

mPIPE: The network engine used to use the TILE_Gx processor. It is a packet processing engine that can support packet header parsing, packet distribution, packet buffer management, load balancing, L4 checksum, packet collection, and packet transmission. Since the packet processing engine is a hardware engine, it is processed at a high speed. Since the API is provided, a programmer can control it programmatically using the API.

TRIO: Transaction is an inseparable processing unit used in the system. TRIO refers to input and output operations performed through transactions. And TRIO controls data movement and buffer management of DMA between tile memory system and PCIe.

PCIe Driver: PCIe is a serial interface for input and output. And twice as much as the existing speed while using existing PCI programming concepts and communication standards [6]. The PCIe Driver dynamically manages the size of the packet buffer.

B. Operation process

Manuals are provided for distributing to 16 cores with the functions provided in TILE-Gx. We have expanded from 16 cores to 28 cores.

Figure 2 shows the process of how packets are handled in the proposed method. When a packet enters mPIPE, it enters packet processing threads (7 core). mPIPE checks only the 5-tuple information and passes it to the PCIe thread (28 core). This is called by-pass. The PCIe thread is immediately passed to TRIO and TRIO transfers to the PCIe driver of the host server. At this time, 28 threads in the PCIe thread are sent to 28 queues. Then, the PCIe driver of the host server receives 28 queues and divides them into 28 channels and handles them to

the core of the host server. This is the fast way to move from TILE-side to Host-side. This is the whole flow of data.

From a hardware point of view, the data passed through XAUI comes into mPIPE. The mPIPE then divides the queue and stores it in memory. Because TRIO uses DMA, it transfers data directly from physical memory. This is the way onto transfer data to the host server without going through packet processing. The speed is extremely fast because there is no packet processing.

The proposed NIC is called MDS-4036GEBT.

C. SDNIC

SDNIC (Software Defined Network Interface Card) is used for performance testing.

The SDNIC application serves as reference code that utilizes a fraction of the cores while delivering line-rate lossless flow and packet processing functionality targeted for network security, network monitoring and optimization and flow-processing offload solutions.

SDNIC supports the following functions.

- Packet header parsing for protocols: VLAN, IPv4, TCP, and UDP
- Flow identification and processing based on IPv4 5-tuple
- APIs to add user-defined flow processing hooks
- Methods to maintain statistics per flow
- Packet filtering (aka classification) based on IPv4 5-tuple with wildcard matching

SDNIC consists of the following components.

- mPIPE classifier: Software running on the Multicore Programmable Intelligent Packet Engine (mPIPE) classifier engine, which load-balances incoming packets onto TILE-Gx cores.
- TILE-Gx-side Software: User-space software running on TILE-Gx, which transfers data between the network and the host memory over the PCIe bus and performs various packet processing functions.
- PCIe host driver: User-space software running on the host, which is a PCIe device driver for the TILE-Encore-Gx36 Intelligent Application Adapter and offers user-space APIs to send and receive data.
- Host-side Application: User-space software running on the host that is the ultimate consumer or producer of network data. The host application operates in polling mode.

IV. TEST RESULTS

TABLE 1. TESTING ENVIRONMENT

	Operating system	Development tools
Host system	Linux CentOS6(64bit)	TileraMDE (4.2.1 version)
MDS-4036GEBT	Linux	
MDS Board	Linux	
Instrument		AX/4000 (10G, 2 port)

Table 1 shows the environment for testing the data transmission and transmission speed of the MDS-4036GEBT.

A. I/O port

To test whether the MDS-4036GEBT is implemented properly, testing four ports operate normally is necessary.

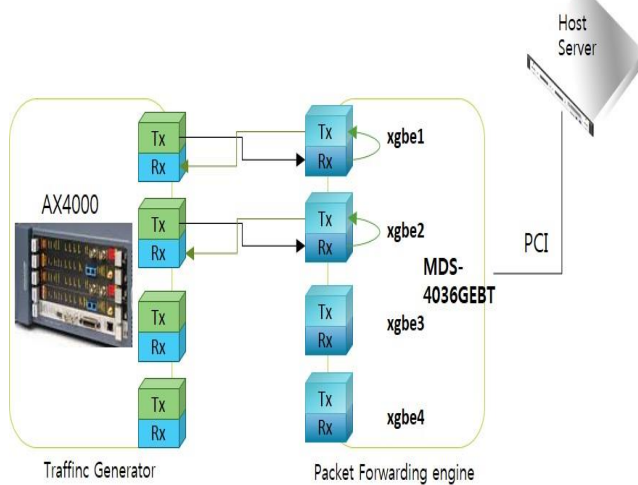


Figure 3. I/O port test implementation diagram

As shown in Figure 3, first connect the testing device to the MDS-4036GEBT with an optical cable. Then prepare the traffic generator and analyzer of the testing device, and then run the packet forwarding engine using the SNIC application of the system. When the testing device generates a packet of each port, the packet is forwarded to the MDS-4036GEBT and forwarded to the testing device.

Figure 4 shows the result of a packet sent from the testing device that can be sent and received normally. The result shows that all four ports of the MDS-4036GEBT have the same number of packets and packet bytes generated by the testing device so that each port operates normally.

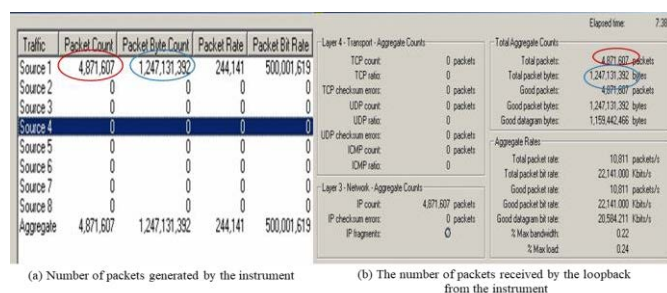


Figure 4. I/O Test results

B. I/O processing performance

Figure 5 shows the configuration of the MDS-4036GEBT to test the I/O processing performance. The MDS board is the device to be used for the performance verification of the MDS-4036GEBT. As shown in the figure, each port of MDS-4036GEBT and MDS board is connected with the 10G interface (XGBE) using an optical cable, and traffic is

generated and monitoring is performed in each device to check the performance.

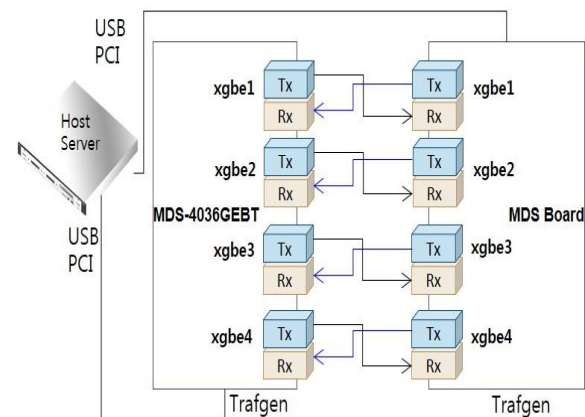


Figure 5. I/O processing performance test implementation diagram

LINK	Tx pkt/s	Tx bits/s	Rx pkt/s	Rx bits/s	Tx err/s	Rx err/s
xgbe1	4644605	9809387088	4590449	9695031400	0	0
xgbe2	4732269	9994553664	4728654	9986917976	0	0
xgbe3	4732209	9994429024	4732209	9994425120	0	0
xgbe4	4732172	9994347056	4728816	9987257824	0	0
xgbe1	4645045	9810350264	4603663	9722935752	0	0
xgbe2	4738016	10006689776	4737432	10005456200	0	0
xgbe3	4737960	10006571936	4737983	10006622216	0	0
xgbe4	4737953	10006559168	4737372	10005329952	0	0
xgbe1	4650546	9821951968	4598426	9711877232	0	0
xgbe2	4738284	10007256112	4738064	10006792864	0	0
xgbe3	4738262	10007209792	4738262	10007207848	0	1
xgbe4	4738255	10007197000	4737903	10006451744	0	0

Figure 6. MDS-4036GEBT monitoring result

LINK	Tx pkt/s	Tx bits/s	Rx pkt/s	Rx bits/s	Tx err/s	Rx err/s
xgbe1	4596616	9708059648	4642226	9804380992	0	0
xgbe2	4732838	9995756144	4732983	9996060128	0	0
xgbe3	4732916	9995924616	4732882	9995847032	0	0
xgbe4	4732773	9995611200	4732270	9994557728	0	0
xgbe1	4603362	9722291224	4647316	9815134216	0	0
xgbe2	4738095	10006855800	4738678	10008089216	0	0
xgbe3	4738651	10008030528	4738661	10008053488	0	0
xgbe4	4738040	10006747984	4738654	10008038776	0	0
xgbe1	4598201	9711395504	4652482	9826041200	0	0
xgbe2	4738933	10008626912	4739152	10009090168	0	0
xgbe3	4739123	10009029104	4739121	10009024888	0	0
xgbe4	4738767	10008277368	4739118	10009018576	0	0
xgbe1	4597441	9709814328	4656893	9835358688	0	0
xgbe2	4738789	10008323696	4739188	10009165968	0	0
xgbe3	4739161	10009109120	4739170	10009128064	0	0
xgbe4	4738417	10007538280	4739177	10009142808	0	0

Figure 7. MDS Board monitoring result

Figure 6-7 shows that the MDS-4036GEBT and MDS board monitor transmit/receive traffic to 4 ports (Tx, Rx) at 10Gbps interface respectively.

V. CONCLUSIONS

In this paper, we propose a high-speed data processing method that by-passes data received through mPIPE and transmits it directly to Host Server through TRIO.

The proposed hardware configuration is implemented in the MDS-4036GEBT. This equipment was used to test the reliability and speed of the proposed method. Traffic generator

and MDS board were used as experimental equipment. Experimental results show that 40Gbps data is transmitted to the host server without dropping.

As the network traffic increases, the efficient use of hardware is one of the important research issues. Therefore, there are a need for continuous research on more efficient and faster hardware configuration methods.

ACKNOWLEDGMENT

This work was supported by “Human Resources Program in Energy Technology” of the Korea Institute of Energy Technology Evaluation and Planning (KETEP), granted financial resource from the Ministry of Trade, Industry & Energy, Republic of Korea. (No. 20164030201330).

This research was supported by the MOTIE(Ministry of Trade, Industry & Energy) and the KIAT(Korea Institute for Advancement of Technology) through N0001132.

*corresponding author is S.G. Choi (choisg@cbnu.ac.kr)

REFERENCES

- [1] Shinji Nishimura, Kazunori Shinoda, Yong Lee, Fumio Yuki, Takashi Takemoto, Hiroki Yamashita, Shinji Tsuji, Masaaki Nido, Masahiko Namiwaka, Taro Kaneko, Kazuhiko Kurata, Shigeyuki Yanagimachi, Naoya Ikeda, “Optical Interconnection for High-speed Routers”, *2011 Optical Fiber Communication Conference and Exposition and the National Fiber Optic Engineers Conference*, pp. 1 – 3, March 2011
- [2] Muhammad Bilal Anwer, Murtaza Motiwala, Mukarram bin Tariq, Nick Feamster, “SwitchBlade: A Platform for Rapid Deployment of Network Protocols on Programmable Hardware”, *SIGCOMM '10 Proceedings of the ACM SIGCOMM 2010 conference*, pp 183 – 194, August. 2010
- [3] Xiaodan Zhang, Jing Xu, Weiping Liu, Hongbin Huang, “The Application Scheme of 40G and 100G Ethernet”, *10th International Conference on Optical Communications and Networks(ICOCN 2011)*, pp.1 – 2, Nov. 2011
- [4] Xingwei Wang, Yinchu Sun, “A QoS Guaranteed Technique for Cloud Applications Based on Software Defined Networking”, *IEEE Access*, Vol. 5, pp. 21229 – 21241, September. 2017
- [5] *TILE-Gx36 Processor(PB033 Rev 3.0)*, Mellanox, 2015-2016
- [6] (2017) Wikipedia website. [Online]. Available: www.wikipedia.org/



Joon Yeol Yoon is currently a B.S. & M.S. candidate in School of Electrical & Computer Engineering, Chungbuk National University, Korea in 2017. His research interest is Network Security.



Won Seok Choi received B.S. and Ph.D. degree in the College of Electrical and Computer Engineering, Chungbuk National University, Korea in 2008 and 2014 respectively. He is Senior Researcher in Fisys Inc. His research interests include Energy saving network, SDN, NFV and NGN.



Jong Oh Kim received B.S. and M.S. degree in Electronics Engineering from Kyeongbuk National University in 1990 and 1992 respectively. He is currently CEO in Fisys Inc. His research interests is SDN, NFV.



Jae Woo Ahn received B.S. degree in the College of Electrical & Computer Engineering, Chungbuk National University, Korea in 2017. He is currently an M.S. candidate in School of Electrical & Computer Engineering, Chungbuk National University. His research interest is network programming.



Jong Beom Kim received B.S. degree in the College of Electrical & Computer Engineering, Chungbuk National University, Korea in 2017. He is currently an M.S. candidate in School of Electrical & Computer Engineering, Chungbuk National University. His research interest is network programming.



Seong Gon Choi received B.S. degree in Electronics Engineering from Kyeongbuk National University in 1990, and M.S. and a Ph.D. degree from KAIST, Korea in 1999 and 2004, respectively. He is currently a professor in the College of Electrical & Computer Engineering, Chungbuk National University. His research interests include smart grid, IoT, mobile communication, high-speed network architecture, and protocol.