Ethernet-based Communication Architecture Design and Fault-Tolerant System

Hyunshin Lee, Hamid Jabbar, Sungju Lee, Saejeong Choi, Quanjie Lie, Inyoung Kim,
Seunghwan Choi, Dongchul Park, Sooyoung Min, Yunsik Lee, and Taikyeong Jeong

***Abstract*— The designs of Ethernet based communication architectures are moving toward the integration of a fault-recovery and fault-detection algorithm on hardware. Each port on the same network-interface card (NIC) design is required to provide highly scalable and low latency communication. In this paper, we present a study of the dual-port architecture and performance of COTS (Commercial Of The Shelf) NIC design which is mainly used in security-enhanced application such as military, finance, automotive and aerospace, in other words: safety-critical applications.**

***Index Terms*— Fault tolerant, Fault-detection, Fault-recovery, Ethernet-based communication**

I. Introduction

Fault tolerant system guarantees availability and reliability in network connections. Redundant network in Fault tolerant configuration allows the user to maintain persistent sessions during a hardware failure or a routing outage or change.

Fault tolerant network interface is new breed of Intelligent Network Interface Card (I-NIC) [1]. Different redundant network interface hardware is available for fault tolerant configuration; this paper discusses the dual-port Network Interface Card (NIC) with the embedded algorithm for high-performance and safety-critical systems. The hardware uses the dual Ethernet ports to handle the hardware fault or routing changes. To achieve high speed data rates the Fault-tolerant NIC employ processor to offload the processing load from the host processor.

Manuscript received January 12, 2012.

Hyunshin Lee is with the Myongji University, South Korea. Email:oshyuns@gmail.com

Hamid Jabbar is with the Myongji University, South Korea. Email:hamid_jabbar@yahoo.com

Sungju Lee is with the Korea University, South Korea. Email:peacfeel7@hotmail.com

Saejeong Choi is with the Myongji University, South Korea. Email:freesaejeong@gmail.com

Quanjie Lie is with the Myongji University, South Korea. Email:joseph@mju.ac.kr

Inyoung Kim is with the Myongji University, South Korea. Email:in880419@naver.com

Seunghwan Choi is with the Myongji University, South Korea. Email:kepchoi@kepri.re.kr

Dongchul Park is with the Myongji University, South Korea. Email: parkd@mu.ac.kr

Sooyoung Min is with the Myongji University, South Korea. Email:minsy@keti.re.kr

Yunsik Lee is with the System IC Design Group, KETI, South Korea. Email:leeys@keti.re.kr

Taikyeong Jeong is with the Myongji University, South Korea. Email:ttjeong@mju.ac.kr

Multiple ports systems are preferable in mission-critical environment. The embedded algorithm allows for fault detection based on both hardware and software faults. Redundant PHY provides failover to a redundant network path. This type of Failover technique is fast as failures can be detected at the physical link layer and by embedded algorithm a new route or tree is pre-located when a failure occurs.

An Intelligent NIC to support multimedia application, having high performance processors along with software architecture have been developed [6]. The applications of fault tolerant networks are unlimited from server applications, financial transaction systems, data transmission in consumer electronics (IPTV etc.), to military systems.

This paper outlines the faults (Section-II), fault testing (Section-V), and hardware requirements for developing a dual-port Fault tolerant Network Interface Card for data speed up to 1Gbps. An embedded processor based NIC with multiple port based on COTS components (Section-IV) is expected to achieve the recovery timing (Section-III) and availability requirement in mission critical networks.

II. Fault Tolerant

Faults or problem causes are not always known, the solution to rectify fault involves both hardware and software as the cause can be of any of the both. Network cable tester is a hardware used for cable fault testing, diagnostic and performance, a laptop is another handy tool with proper TCP/IP software's installed; Ping is commonly used command to test the network. But these solutions involve human, are time consuming and not applicable in mission critical networks.

A. Fault Tolerant Network

Fault-tolerant Network or "high availability" network systems describes a computer system or network interface designed in such a manner that in a case a component fails or software error occurs, a backup or redundant component or procedure can immediately take its place (recovery) with no loss of service in min. time, without human intervention [7].

Fault tolerance capability can be attained in software, hardware or both. In Fault-Tolerant networks, usually the hardware, network Switch, Network Interface Card (NIC), Ethernet ports are duplex. Data loss and fault recovery times can vary with the fault tolerant technique and hardware.

B. Network Faults

It becomes hard to list all faults but can be generalized in hardware and software faults. Software faults can range from software error or bug either in OS or embedded software of switch or router. A large number of data transactions which

ISBN: 978-988-19251-9-0
ISSN: 2078-0958 (Print); ISSN: 2078-0966 (Online)

processor cannot handle can cause a temporary outage of communication.

Hardware faults can be malfunction of hardware in client or any other component (switch, router, cable) in the network system or path.

C. *Fault Injection for Tolerance System*

Existing systems are designed to be tolerant to only certain faults, in certain conditions; tradeoff exists between fault types and combinations [9].

In general, fault injection technology is a way to investigate the fault tolerance system. Different hardware and software based fault injection techniques are available to validate the performance of the fault tolerant system. A Fault and Error Automatic Real-time Injector (FERRARI) system is developed to inject transient errors and permanent faults for concurrent error detection and correction [2].

The NIC hardware design presented is capable to handle the intentional dummy faults, injected for testing, and also to handle the large amount of traffic generated for fault injection, testing and keeping in view the real world scenario.

III. Proposed System

Different fault-tolerant related products, especially architecture of DP-FT-NIC (Dual-port Fault-Tolerance-NIC) available in market were studied and analyzed thoroughly for development of DP-FT-NIC to match our performance needs, the time for minimum replacement and when the spares run out.

It was observed that the COTS hardware is available to support gigabit Ethernet, PCI/PCI-X, and processing power. The software platform especially the algorithm and associated test benches are not developed for embedded faults tolerant NIC. The System is designed in a modular block style, to replace it with new components of better performance, even though NIC can be replaced.

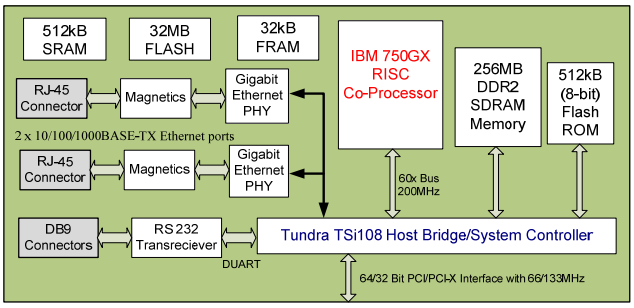

Fig. 1. Block Diagram of IBM 750GX Evaluation Board

To evaluate the performance of the fault tolerant system, it was desired to have an evaluation kit with the required hardware and software capabilities to handle the high data transfer rate and processing power along with software development tools and performance analyzers.

IBM PowerPC 750GX/GL evaluation board, having IBM PPC750GX processor and Tundra TSI108 host bridge with supporting circuits and components as shown in Fig. 1 was selected for the said purpose. The PPC 750GX/GL is targeted for high performance, low power systems (2.5W typical, 3.7W max) that use a 60x bus.

The tool chain, IBM embedded Power PC Operating system (EPOS), PowerPC Initialization Boot Software (PIBS), benchmarks tools [16] provided the entire necessary software platform to design, test, and evaluate the Fault-Tolerant algorithm with Dual port Ethernet as shown in Fig. 2.

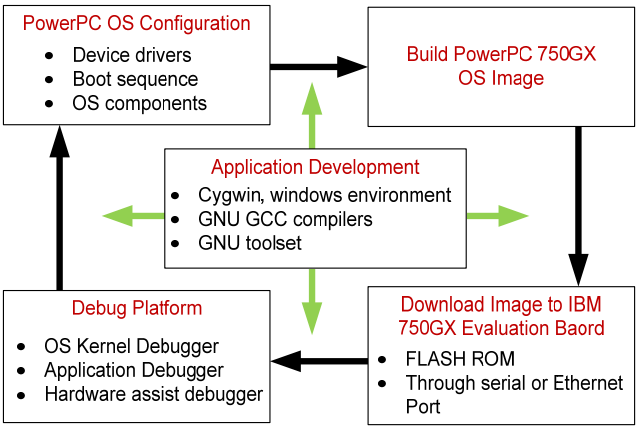

Fig. 2. Block diagram of the designed Dual Port Network interface card with embedded processor

For Cross development, GNU toolset were build in windows-XP using cygwin which is a linux-window converting utility. The EPOS consists of function that are platform independent such as kernel, TCP/IP protocol stack, shell command interpreter and a PCI device manager using API (Application Programming Interface) calls.

IV. ALGORITHM DESIGN

The challenge lies in design of fault tolerant software to support the COTS hardware and complexity. Using the Evaluation kit and different scenarios we were able to device an algorithm at higher level for the network consisting of Dual port fault-tolerant NIC as shown in Fig. 3, which can be extended for multiple port FT-NIC. Each client has DP-FT-NIC, two networks via switch A and B builds redundancy among Client-1 and Client-2 and 3. We are investigated by the 3 steps: self-awareness, network-awareness and time-stamping.

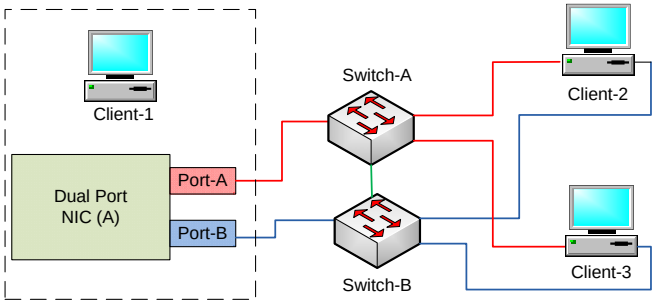

Fig. 3. Basic connection scheme for proposed algorithm

A. *Self- Awareness*

This section checks the connectivity between the two ports of the same Dual-port FT-NIC (DP-FT-NIC). It is necessary to evaluate the communication of the both Ethernet ports to

ISBN: 978-988-19251-9-0
ISSN: 2078-0958 (Print); ISSN: 2078-0966 (Online)

be aware of any faults caused by lose connector, broken wire, no connection etc.

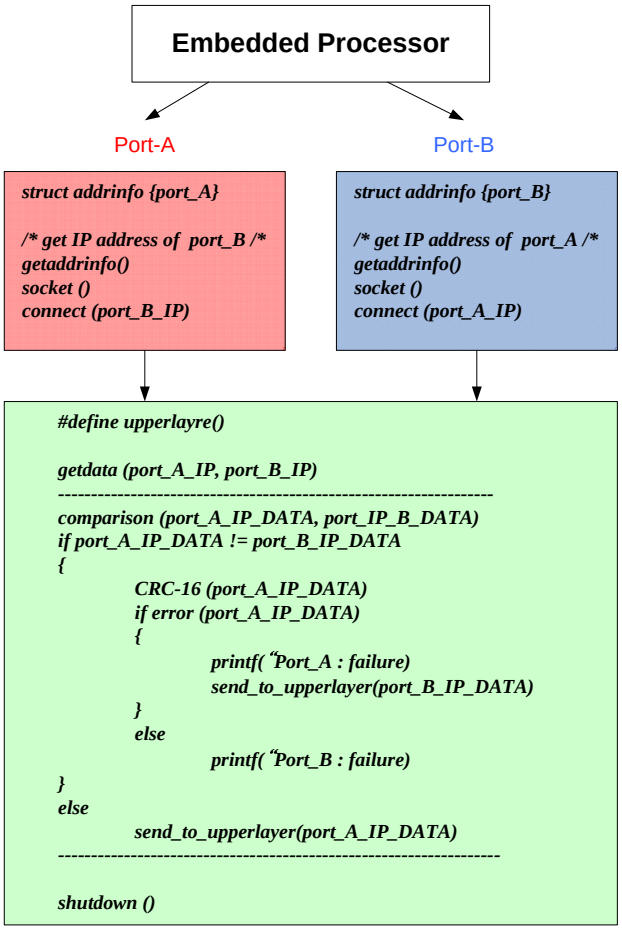

Fig. 4. Self-awareness scheme API level code

The EPOS simultaneously receives data through Port A and Port B as shown in Fig. . To detect a fault, this system compare with each other's data. If two data have a difference, the CRC-16 method for Port A's data is performed. The error of CRC-16 method means that Port A has a failure. The comparison and the restricted CRC-16 achieve reliability and a high speed.

B. Time-Stamping

Timestamps exchanged are used to determine individual roundtrip delays and clock offsets, as well as provide reliable error estimates. Clock differences such as local-clock resolution and skew error need to be minimized; techniques such as [11] can be used for such purposes.

In practice, errors due to stochastic network delays dominate, however, it is not usually possible to characterize network delays as a stationary random process, since network queues can grow and shrink in chaotic fashion and arriving customer traffic is frequently bursty.

As shown in Fig. 5, a client broadcast a package to be received by all other clients, the reply contains the time information of that client. The receive packets from all the clients are used for computing the time of the day, delay, clock resolution etc.

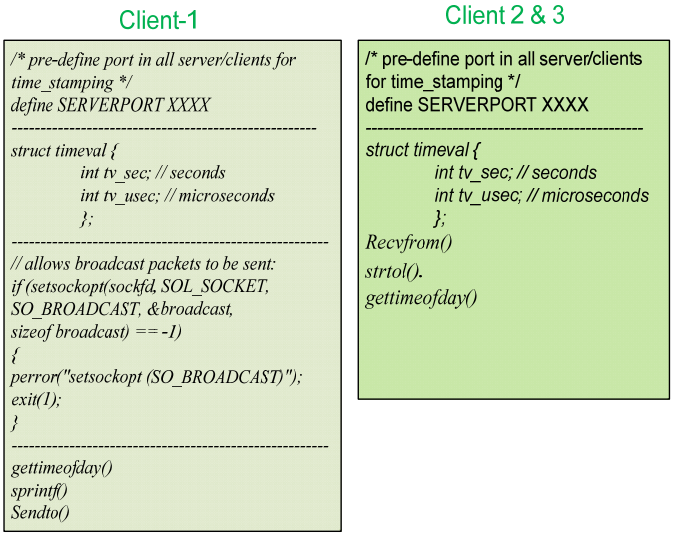

Fig. 5. Time Stampping scheme API level code

V. Simulation

In order to evaluate our approaches for higher reliable system, we make a software program by using server-client model, and test using reality data set. Although, this program was very simply implemented, it can provide higher reliable than straightforward server-client program. This program consists of two operations by using the reconnection and the switching two ports. For that, first, we can disconnect the network connection during the transmission data. Secondly, when some errors occurred in a port, it exchanges the two ports. Also, for the detection of the errors, we used the CRC-16 method, and it was performed at once, when the difference exists in compared with two ports. Especially, since CRC-16 was performed at once for the difference of the transmission data, it can be more efficiently performed in run-time than simple one. Figure 6 shows our implemented software program with server-client system.

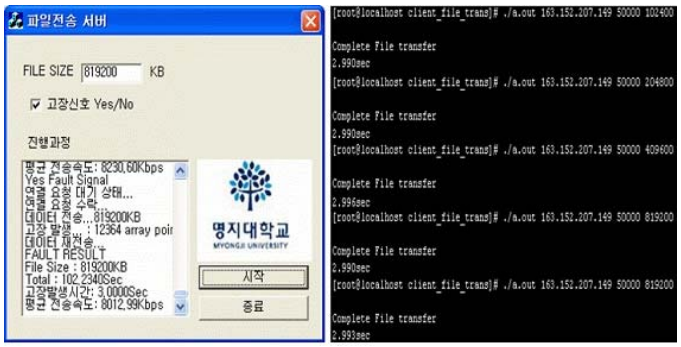

Figure 6: (a) A screenshot of File Transfer Server on the designed program, (b) Booting sequences with CRC-16 check

Finally, we confirmed the performance for CRC-16 method. Typically, with using CRC-16, we can guarantee the 16 byte data integrity during the transmission with using the 2 byte. In our system, the average time for 1Kbyte integrity was 28ms. Also, since CRC-16 was applied at once for the difference of the transmission data, it can more efficiently be

ISBN: 978-988-19251-9-0
ISSN: 2078-0958 (Print); ISSN: 2078-0966 (Online)

performed in run time than simple applying. Therefore, our system can provide both higher reliable and speed in run-time.

VI. Port Configuration

Physical (PHY) layer provides the interface between the media access control (MAC) layer and the tarnsrecievers in gigabit Ethernet interface.

The 1000BASE-X physical layer, also referred to as the GbE physical layer, consists of three major blocks, the Physical Coding Sublayer (PCS), the Physical Medium Attachment sublayer (PMA), and the Physical Medium Dependent sublayer (PMD).

Giga bit Ethernet uses all four cable pairs for simultaneous transmission in both directions.

The PCS transmit and receive sections communicate with the Physical Media independent or dependent sublayers to transmit and receive data via magnetic and RJ-45 connector. Magnetics are like transformer and are required at high data transmission rate. High performance Analog to Digital Converter (A/D) and Digital to Analog Converter (D/A) are needed to detect the cable faults using the time-domain reflectometry (TDR).

VII. Related Works

Development of fault tolerant system is closely related to fault testing techniques. Analytical modeling, experimental techniques, fault and error injections techniques through which statistical parameters can be obtained for improvement and limitation analysis are already being developed and tested.

CERN's LHC accelerator used giga-bit Ethernet and PCI based INIC for error free data acquisition [8]. The criticality of data communication in above example scenarios as well as in other seen or unseen future applications, Fault Tolerant Network Interface Card promises reliable communication in multiple failure scenarios.

Using commonly available hardware and software, a fault-tolerant solution is developed for IP over Ethernet networks with no need for modification of existing networking equipment [9]. A middleware-based fault-tolerant Ethernet is developed for process control networks with no change to commercial of-the-shelf hardware and software and is transparent to IP-based applications achieving less than 1-ms end-to-end swap time and less than 2-sec failover time [4].

Multiple port network interface cards with the embedded processor, PCI port and data storage capabilities are being developed by different companies [12] [13], but still the existing embedded software needs to be upgraded to achieve the optimum fault tolerance capabilities.

VIII. Conclusion

This paper defines the timing matrix required for fault detection and recovery, this put the constraints on the hardware required and embedded algorithm.

On basis of the testing using the reference hardware we were able to design a dual port NIC which can achieve the fault tolerance for mission critical systems. The hardware design is capable to handle Giga bit Ethernet communication and heavy workload.

1) We have also defined the algorithm which can be embedded in the processor to achieve fault tolerance along with the data communication services which is the prime task for NIC. The new algorithm is simulated using the CRC. API driven algorithm, benchmarks and related verification provide comparable results to synthetic workloads, but higher performance can be achieved with specific low level programming.

Acknowledgment

This work was supported by the IT R&D program of The MKE/KEIT. 10040191, the development of Automotive Synchronous Ethernet combined IVN/OVN and safety control system for 1Gbps class. This work is supported by the Korea gov. Ministry of Knowledge and Economics(MKE) under the grant No. I-2010-1-012 of the Electric Power Industry Tech. Evaluation and Planning Center (ETEP). This work is supported by National Science Foundation (NRF) grant funded by the Korea gov. (MEST) (No.20090069991). The circuit was designed at IC Design Center.

References

[1] M. H. Davis Jr., "An Intelligent Network Interface Card," Proceedings of 18th Digital Avionics Systems Conference, USA, vol. 2, pp.9.D.1-1-9.D.1-6, 1999.

[2] FERRARI: A flexible software-based fault and Error Injection System, IEEE transactions on Computer, 2006.

[3] D. Song, B. Jang, C. Hoon Lee, "Design and implementation of fault tolerant communication middleware for a high reliable launch control system".

[4] S. Song, J. Huang, P. Kappler, R. Freimark, and T. Kozlik, "Fault-tolerant Ethernet middleware for IP-based process control networks," lcn, pp.116, 25th Annual IEEE International Conference on Local Computer Networks (LCN'00), 2000.

[5] J. Huang, S. Song, L. Li, P. Kappler, R. Freimark, J. Gustin, and T. Kozlik, "An open solution to fault-tolerant Ethernet: design, prototyping, and evaluation," IEEE International Performance, Computing and Communications Conference, 1999. IPCCC '99, Scottsdale, AZ, USA, pp. 461-468, 10-12 Feb, 1999.

[6] W. J. Dally, and B. Towels, "Principles and practices of Interconnection Networks," Elsevier, Morgan Kaufmann Publishers, 2004. http://www.mkp.com/companions/0122007514.

[7] J. Baranski, M. Crocker, and G. Lazarou, "Fault-tolerant Reconfigurable Ethernet-based IP Network Proxy", proceeding of Communications, Internet, and Information Technology, Nov. 17–19, 2003 Scottsdale, AZ, USA.

[8] GE Fanuc, Intelligent Platforms, Ethernet Interface products, *Online http://www.gefanuc.com*.

[9] Xpedite4002, high-performance Processor PMC Module, Extreme Engineering Solutions, *Online http://www.xes-inc.com*.

[10] Honeywell, Fault Tolerant Ethernet (FTE) Products. *Online http://www.honeywell.com*.

[11] User's Manual, PowerPc 750GX/GL Evaluation Board, IBM, *Online https://www-01.ibm.com/chips/techlib/techlib.nsf/products/PowerPC_750GX*
GL_Evaluation_Kit, Tsi108, Host Bridhe, Tundra, *Online http://www.tundra.com/products/host-bridges/tsi108*.

[12] H. Jabbar, and T. Jeong, "Fault Tolent Network Interface Card Design for Combat System Network (CSN)," Proceedings of ICDC 2008, Japan, May 2008.

[13] H. Jabbar, S. Kim, D. Lee, Y. Ryu, and T. Jeong, "PCI Based Dual-port Network Interface Card Design and Testing for Mission Critical System," Proceeding of 12th CI Seminar in Korea, 2008.