한국통신학회논문지 '12-11 Vol.37C No.11
http://dx.doi.org/10.7840/kics.2012.37C.11.1162

단일 MAC을 이용한 자동 고장 극복 Ethernet NIC (Network Interface Card) 장치 구현

김 세 목◆, PHAM Hoang-Anh*, 이 동 호**, 이 종 명°

A Novel Implementation of Fault-Tolerant Ethernet NIC (Network Interface Card) Using Single MAC

Se-Mog Kim◆ Hoang-Anh Pham*, Dong-Ho Lee**, Jong Myung Rhee°

요 약

Mission critical 임무를 수행하는 Ethernet 기반 첨단 네트워크 시스템에서 자동 고장 극복 기능은 시스템의 중단 없는 운용을 위한 중요한 요구사항 중의 하나이다. 이러한 고장 극복 기능은 네트워크 시스템의 각 노드에 멀티 port를 지원하는 네트워크 인터페이스 카드 (Network Interface Card, NIC)를 설치함으로써 가능하다. 현재 가용한 NIC 장치는 두 개 또는 그 이상의 MAC (Media Access Control)을 사용하여, active port 고장 시에 MAC switching하여 자동 고장 극복 기능을 수행한다. 이러한 NIC 장치는 일반적으로 co-processor 및 이를 위한 펌웨어 (firmware)를 필요로 하며, 이에 따라 고장 극복 시간이 길어지고 throughput이 저하되는 단점이 있다. 또한 co-processor를 위한 펌웨어는 전술 환경 변화에 따라 upgrade를 해야 하므로 고장 극복 장치 가격도 상승하게 한다.

본 논문은 기존 하드웨어 방식에서 일반적으로 사용하는 co-processor와 다수 MAC 대신에, 하나의 MAC 만을 사용하는 새로운 하드웨어 방식 NIC 장치 설계 방안을 제시한다. 제시된 새로운 NIC은 단일 MAC과 일반 로직 게이트 블럭으로 설계하여 고장 극복 기능을 수행한다. 제안 방식에 따라 NIC을 구현하여 성능 실험을 통해 기존 방식 대비 우수함을 입증하였다.

Key Words : FTE, Fail-over time, Single MAC, NIC

ABSTRACT

One of the important operational requirements for mission critical Ethernet networked system is having the fault tolerant capability. Such capability can be obtained by equipping multiport Network Interface Card (NIC) in each node in the system. Conventional NIC uses two or more Media Access Controls (MACs) and a co-processor for the MAC switching whenever an active port fails. Since firmware is needed for the co-processor, longer fail-over switching and degraded throughput can be generally expected. Furthermore the system upgrading requiring the firmware revision in each tactical node demands high cost. In this paper we propose a novel single MAC based NIC that does not use a co-processor, but just use general discrete building blocks such as MAC chip and switching chip, which results in better performances than conventional method. Experimental results validate our scheme.

※ 본 연구는 국방과학연구소와 방위사업청 과제(RD100004KD) 지원으로 수행되었습니다.
◆ 주저자 : 명지대학교 정보통신공학과 Ubiquitous&Convergence 연구실, semog@mju.ac.kr, 준회원
° 교신저자 : 명지대학교 정보통신공학과, jmr77@mju.ac.kr, 종신회원,
* 명지대학교 정보통신공학과 Ubiquitous&Convergence 연구실, anhph@mju.ac.kr, 준회원
** 국방과학연구소, hangboknara@add.re.kr
논문번호 : KIC2012-08-372, 접수일자 : 2012년 08월 27일, 최종논문접수일자 : 2012년 11월 9일

Ⅰ. 서 론

고장 극복 (Fault Tolerant) 기능은 네트워크 기반 첨단 임무 체계 (예: NCW, 함정 전투체계 네트워크, 무인기, 전차, 유도 무기, C4I자동화 체계 등)에서 중요한 시스템 요구사항 중의 하나이다. 이러한 체계에서는 고성능은 물론, 높은 신뢰도와 생존성이 절대적으로 요구된다. 그러므로 체계 신뢰도를 높이기 위하여, 고장이 발생하여도 지속적으로 시스템이 운영되는 고장 극복 기능 확보가 필수적이다[1]. 이러한 첨단 임무 체계에 사용되는 네트워크는 Ethernet이 De facto Standard로 되어 오랫동안 시장을 형성하고 있으나, 고장 극복(Fault-Tolerant) 기능은 표준에 포함되어 제공되지 않는 실정이다. 따라서 다양한 방식의 Fault Tolerant Ethernet (FTE)이 연구 개발되었으며, 기본적으로 소프트웨어 방식과 하드웨어 방식으로 구분된다[2-8].

소프트웨어 방식은 네트워크에 속한 각 노드가 자신을 제외한 모든 노드에 자신의 존재를 알리는 하트비트 (Heart-beat) 신호를 주기적으로 전송하고, 하트비트 신호를 받지 못하는 port를 고장으로 간주하여 하트비트 신호가 안정적으로 유지되는 port를 선택하여 사용하도록 하는 방식이다. 즉, 각 노드는 두 개 port, 주 port(Primary port)와 예비 port (Standby port)로 구성된다. 주 port로 통신을 하던 중, 주 port가 하트비트 신호를 수신하지 못하면 해당 port의 고장을 선언하고 예비 port로 전환 (switching)하여 통신을 유지한다. 만약 예비 port에서도 하트비트 신호를 수신하지 못하였다면, 해당 노드는 통신 단절 상태로 인식한다. 이러한 소프트웨어 방식은 특정한 하드웨어가 아닌 범용 하드웨어인 Commercial-Off-The-Shelf (COTS) 장치를 그대로 사용하는 장점이 있다. 그러나 하트비트 송/수신에 따른 네트워크 부하가 가중됨은 물론, 소프트웨어를 통하여 고장을 판단하는 방식이므로 고장 극복 시간은 하드웨어 방식에 비해 상대적으로 길다 (통상적으로 1초 정도)[5].

하드웨어 방식은 각 노드에 구성되어 있는 Network Interface card (NIC) 장치의 두 개 port에서 물리적인 신호를 확인하여 고장 극복 기능을 지원하는 방식이다. 즉, 노드의 port에서 물리적인 신호가 없는 port를 고장으로 간주하고 물리적 신호가 안정적으로 유지하고 있는 port를 선택하여 사용한다. 이 방식은 소프트웨어 방식과 비교시, 네트워크에 추가적인 부하를 주지 않으며 port 고장에 따른 port 전환 시간이 빠른 장점을 가지고 있어 실시간 임무가 필요한 네트워크 기반 체계에 매우 적합한 방식이다. 기존 하드웨어 방식은 FTE 구현을 위해, 두 개 이상의 Media Access Control (MAC)을 Active와 Standby로 설정하여 외부와 통신을 시도하며, 내장된 Co-processor와 내부 MAC에 의하여 active MAC에서 고장 발생시 standby MAC을 선택하도록 한다[8]. 이 방식은 안정적으로 standby port를 선택할 수 있지만, 그에 따른 고장 극복 시간이 길어지고 (상용은 수백 msec, 군용은 수십 msec 정도), throughput이 크게 저하될 수 있다. 특히 실시간 대응이 필요한 특수 임무에서 전장 상황을 전달하는 영상 정보 전송 등은 급속히 응용이 증대되고 있으며, 이에 따른 throughput 성능 개선은 필수적으로 요구되고 있다.

소프트웨어 방식과 하드웨어 방식 FTE를 비교하면 Table 1과 같다.

Table 1. Comparison between hardware based FTE and software based FTE

	Hardware based	Software based
Fault verification	physical signal	heart-beat signal
Fault declaration	self	upper layer
FTE application	node level	middleware based system level
Fault effectiveness	single node	system
Switching Speed	fast	slow
COTS usage	difficult	easy

본 논문은 하드웨어 방식 이중화 FTE 장치를 기존 방식의 co-processor와 3개 MAC을 사용하지 않고, logic gate를 이용하여 포트를 전환하도록 단일 MAC을 사용하는 새로운 하드웨어 FTE 장치 설계 방법을 제안한다. 단일 MAC을 사용함으로써, 장치 내부에 추가적인 프로세서를 탑재하지 않아 고장 극복 시간을 개선할 수 있으며, 특히 전체 네트워크 시스템의 통신 성능을 좌우하는 throughput은 획기적으로 개선된다. 또한 프로세서 탑재에 따른 펌웨어 (firmware) 개발 등의 추가 비용도 절감할 수 있다.

본 논문의 구성은 다음과 같다. Ⅱ장에서는 세 개 MAC을 사용한 기존 하드웨어 방식을 설명하고, 기존 하드웨어 방식의 문제점을 해결하기위한 새로운 FTE 설계 방식을 구현 제시한다. Ⅲ장에서는 구현된 장치를 이용하여 개선된 성능을 실험을 통하여 검증한다. 마지막으로 Ⅳ장에서 본 논문의 결론

을 제시한다.

Ⅱ. 본 론

2.1. 기존 방식 FTE 장치

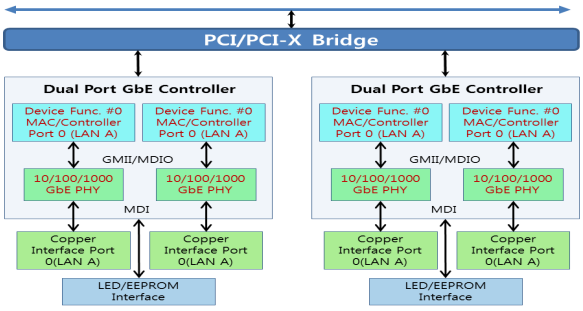
일반적으로 하드웨어 방식 FTE는 네트워크 시스템 단말에 이중 또는 다중 port로 구성된 NIC을 사용한다. 예를 들어 기존 이중화 FTE는 3개 MAC을 사용하고 있다. 즉, 어플리케이션을 위해 사용되는 내부 MAC과 2개 포트에 각각 할당되어 있는 2개 MAC이 있다. 또한 내부 MAC과 연결된 active port MAC에서 일정 시간동안 통신 신호가 입력되지 않으면, 내장된 co-processor에 의해 active port MAC을 고장으로 간주하고, 대기모드로 설정되어 있는 standby port MAC으로 통신 port를 전환 연결하도록 한다. 즉, 기존 하드웨어 방식 FTE 장치에서는 고장 극복 기능을 추가하기 위해 co-processor를 탑재하여 구성하고 있다. co-processor는 전송되는 데이터 신호를 감지하고, port 고장이 발생시 port 전환을 수행하며, 두 개 port에서 사용하는 서로 다른 MAC 주소를 내부 어플리케이션에서 사용하는 MAC 주소와 매칭을 시켜주는 기능을 한다. 또한 co-processor를 사용하기 위해서는 펌웨어를 고장 극복 장치에 탑재한다. Figure 1은 기존 하드웨어 방식 FTE 장치 예이다.

군용 하드웨어 방식 FTE 장치는 고장시 피해를 최소화하기 위하여, 상용 FTE 장치보다 빠른 고장 극복 시간을 갖도록 설계된다[7].

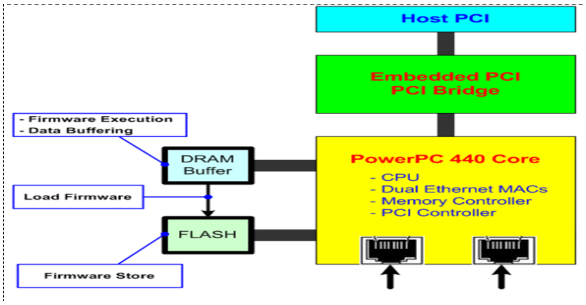
2.2. 기존 하드웨어 방식 FTE 문제점

기존 하드웨어 방식 FTE 장치는 현재 다양한 분야에서 응용되고 있으며, 네트워크 통신 발달에 따라 더욱 확대될 것이다. 이에 따라 FTE 성능도 지속적으로 개선되어야 한다. 그러나 기존 하드웨어 방식 FTE 장치 성능을 개선하기 위해서는 많은 노력이 필요하다.

즉, co-processor 성능을 분석하고, 분석된 성능을 최대한 활용할 수 있는 펌웨어를 개발해야 하며, 펌웨어가 안정적으로 동작하도록 해야 한다. 따라서 펌웨어 개발은 상당한 시간이 소요되며, 또한 통신환경 변화에 맞춰 지속적으로 upgrade되어야 한다. 네트워크 통신환경은 빠르게 변화하고 있으며, FTE 장치 성능 변화 주기도 통신환경 변화속도에 맞추어져야 한다.



a. Functional block diagram of NIC



b. NIC configuration

Fig. 1. Conventional FTE device[8]

그러나 기존 설계 방식으로는 네트워크 통신 발전 속도에 따른 FTE 장치 성능 개선이 매우 힘들기에 새로운 설계방안이 요구되고 있다. 즉, 하드웨어 방식 FTE 장치에서 co-processor를 사용하지 않는 설계가 필요하다. co-processor가 없는 NIC설계에서 고려할 점은 다음과 같다.

① 빠른 포트 전환 및 throughput 특성 최대한 유지
② 간편한 upgrade
③ 보편적 사용위한 저 개발 비용

2.3. 새로운 방식의 FTE 장치

본 논문은 co-processor를 사용하지 않는 새로운 하드웨어 방식 FTE 장치를 제안한다. 제안방식은 logic gate를 이용하여 desecrate한 block을 구성하고 port를 전환하는데, 기본 구성은 Fig. 2와 같다. Fig. 2에 제시한 바와 같이, 외부로 연결되는 두 개 RJ45 단자인 Port0 및 Port1과, 입·출력 신호를 전송하는 PHY0과 PHY1 chip, MAC chip, 그리고 PHY와 MAC을 연결하고 노드에 데이터를 전송하도록 하는 Media Independent Interface (MII)/ Gigabit Media Independent Interface (GMII) 신호로 구성된다. MII/GMII 신호를 switching 하도록 logic gate 기반 switching block을 구성하여, Port0

이 active 상태로 동작하다가 Port0에 고장이 발생하면, standby 상태에 있던 Port1로 통신 port가 전환되고, Port1이 active 상태로 동작하게 된다. 즉, Port0이 active port인 경우, PHY0의 입·출력 신호는 정상적으로 동작을 한다. 이러한 Port0의 active 상태는 MII/GMII 신호를 통하여 switching block에 입력되어, PHY1을 disable시켜 Port1이 standby 상태가 되도록 설정한다. 만약 active port인 Port0의 입·출력 신호(PHY0)에 문제가 발생하여 active port를 나타내는 PHY0의 MII/GMII 신호가 사라지는 경우, 이 정보는 즉시 MII/GMII switching block에 입력되고 PHY1을 구동시켜 standby port인 Port1을 active로 동작하게 한다. Fig. 3은 port 전환 개념이다. 이와 같이 MAC으로 통신 신호가 입력되기 이전인 MII/GMII 신호 체계에서 고장 극복 기능이 구현되기에, port 변화에 따라 MAC 주소가 변경되어 내장된 co-processor가 port의 MAC 주소와 상위 어플리케이션에서 사용하는 MAC 주소를 매칭할 필요가 없다. 따라서 co-processor의 지원 없이 고장 극복 기능을 수행한다. Table 2에 제안 방식과 기존 방식을 비교 제시하였다.

Table 2. Comparison between proposed FTE and conventional FTE

	Proposed	Conventional
Design	logic gate design for switching and single MAC usage	firmware design and three MACs usage
Co-processor	no	yes
Switching speed control	CPU control	logic gate control
System optimization	difficult	easy
Development Cost	low	high

Fig. 4는 제안 방식에 따라 구현한 프로토타입이다. MAC chip은 AX88742 (ASIX사 제품), PHY chip은 AC101QF (Altima communications사 제품)를 사용하였다. 사용한 부품들은 범용으로, 유사한 기능을 보유한 타 부품을 사용하는 경우에도 동일한 설계로 구성 가능하다.

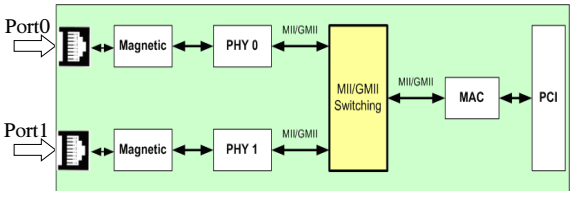


Fig. 2. Block diagram of proposed FTE NIC

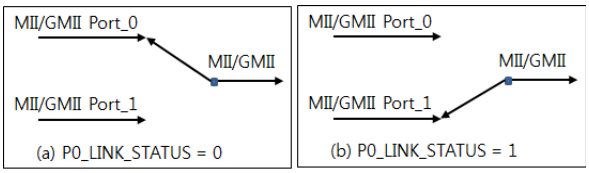


Fig. 3. Concept of port switching

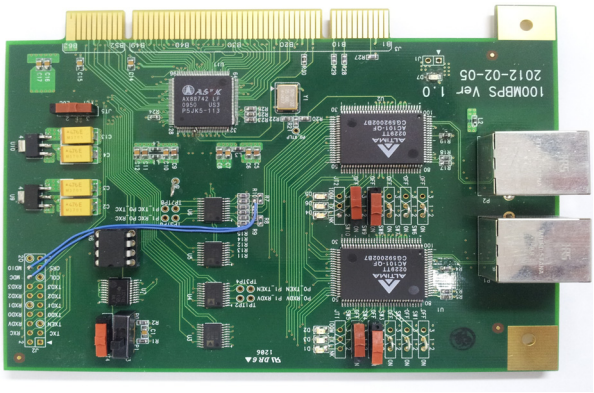


Fig. 4. Proposed NIC implementation

Ⅲ. 실 험

본 장에서는 제안방식으로 구현한 NIC을 이용하여 고장 극복 기능의 주요 성능 요소인 고장 극복 시간과 throughput 실험 및 결과를 제시한다. 고장 극복 시간은 고장 발생 후, port 전환을 통하여 지속적으로 통신이 가능한 상태로 되는 고장 극복 시간만을 측정하고, throughput은 통신 port를 통한 데이터 전송 능력을 측정하여 각각 성능 결과를 제시하였다.

고속으로 port가 전환되는 것을 확인하기 위한 테스트 시스템 구성을 Fig. 5에, 고장 극복 시간을 측정하기 위한 신호 흐름을 Fig. 6에, throughput 측정을 위한 테스트 시스템 구성을 Fig. 8, 그리고 사용한 측정 장비는 Table 3에 나타내었다.

3.1. 고장 극복 시간 측정

Fig. 5의 구성과 측정방법은 RFC-2544 기반 데이터 전송(UDP)을 기준으로 구성한 것이다. Packet Generator에서는 지속적으로 패킷을 발생하여 Packet Receiver로 패킷을 전송한다. Switch에 연결되어 있는 Packet Receiver의 Port_A와 Port_B는 active 상태에 있는 port에서만 패킷을 수신한다. 이러한 수신을 확인하기위해 Packet Monitor를 연결하여 Port_A와 Port_B로 전송되는 패킷 전송 시간을 Switch의 mirroring 기능을 통하여 확인한다. 초

기 Port_A는 Active port로 동작하고, Port_B는 Standby로 동작하도록 구성한 후, Port_A로 연결된 Link를 단절 시킨다. Port_A의 연결이 끊어지고, Port_B가 동작함을 확인하고 port 전환 시간을 측정한다. 고장 극복 시간 측정은 Packet Receiver Port_B에 처음 수신된 패킷 도착 시간과 Port_A에 최종 수신된 패킷 도착 시간 차이로 계산할 수 있다[7,9,10]. Fig. 7에 고장 극복 시간 산출 방법을 나타내었다. Fig. 7에서 Port_A에 마지막 패킷의 도착 시간은 T_K이고, Port_B로 첫 패킷이 도착한 시간을 T_M이라고 하면, 고장 극복 시간, T_{fot}는 다음 식과 같이 나타낼 수 있다.

$$T_{fot} = T_M - T_K \quad (1)$$

고장 극복 시간을 측정하는 절차는 다음과 같다[6,9].

① Fig. 5와 같이 측정 시스템을 구성한다.
② Packet Generator에는 패킷 송신 프로그램, Packet Receiver에는 패킷 수신 프로그램, 그리고 Packet Monitor에는 패킷 캡쳐 프로그램을 설치한다.
③ Packet Generator, Packet Receiver 및 Packet Monitor PC에 설치된 각 프로그램을 실행한다.
④ Packet Generator는 설정한 크기의 패킷을 UDP를 통해 지속적으로 Packet Receiver로 전송한다.
⑤ 데이터 송신 중, Packet Receiver의 active port에 연결된 링크를 단절시킨다(케이블을 Unplug 한다).
⑥ 데이터 송/수신 완료 후, Packet Monitor에서 캡쳐된 패킷 수신 시간을 분석하여 port 전환 시간을 확인한다.
⑦ ③~⑥ 과정을 50회 반복 수행한다.

제시된 절차대로 본 논문에서 제안한 방식으로 구현된 제품과 mission-critical한 시스템에서 주로 사용되는 기존 하드웨어 방식 NIC을 비교 측정하였다. 측정된 고장 극복 시간을 Table 4에 제시하였다. 제안 방식은 기존 방식 대비 동일 또는 우수한 고장 극복 시간 특성을 나타낸다.

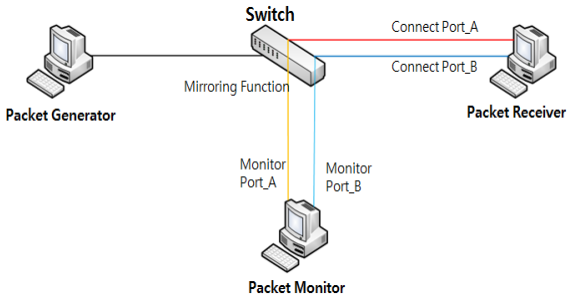


Fig. 5. Test system for fail-over time measurement

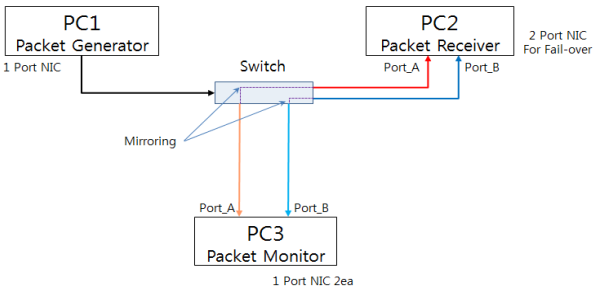


Fig. 6. Signal flow of Figure 5

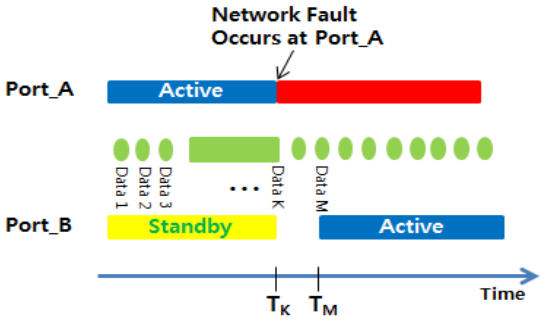


Fig. 7. Methodology for fail-over time measurement

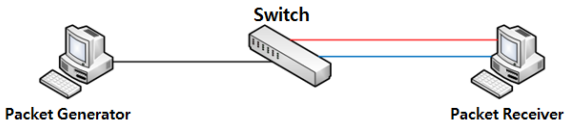


Fig. 8. Test system for throughput measurement

Table 3. Equipment detail for both Test systems

	Detail	OS
Packet Generator	Samsung Laptop, NT-N310	Windows XP
Packet Receiver	Dell PC, 5150	Linux (Ubuntu)
Packet Monitor	PC	Linux (Ubuntu)
Switch	Cisco, Catalyst 3560	

Table 4. Test Results for fail-over time (unit : msec)

		Proposed NIC			Conventional NIC		
		Max.	Min.	Ave.	Max.	Min.	Ave.
Packet Size	256bytes	6	12	9	4	26	11
	512bytes	6	12	10	6	25	10
	1,024bytes	6	27	11	6	29	12

3.2. Throughput 성능 측정

고장 극복용 NIC은 네트워크 단말에 설치되어, 단말에서 외부로 통신하는 모든 통신 패킷을 전달하도록 되어있다. 따라서 구현한 장치를 통과하는 데이터 전송 용량을 나타내는 throughput은 전체 네트워크 시스템의 전송 용량을 결정하는 중요한 요소이다. 다음은 throughput 측정 절차이다.

① Fig. 8과 같이 측정 시스템을 구성한다.

② Packet Generator와 Packet Receiver에 throughput 성능을 평가하기 위한 eTTCP 프로그램을 설치한다[11].

③ Packet Generator에 설치된 eTTCP는 송신모드로, Packet Receiver에 설치된 eTTCP는 수신모드로 프로그램을 실행한다.

④ Packet Generator에 설치된 eTTCP 프로그램에서 단일 전송하는 패킷 크기를 설정하고, 설정된 패킷 크기에 따라 패킷을 생성하여 Packet Receiver로 전송하도록 한다.

⑤ Packet Generator에서 전송되는 패킷과 Packet Receiver에서 수신된 패킷을 비교하여 손실여부를 확인한다.

⑥ 패킷 손실이 발생하였을 경우, 재전송을 시도하여 패킷 손실이 없는 상태에서 단위 시간당 송신 패킷 양을 측정한다.

⑦ ③~⑥번 과정을 50회 반복 수행한다.

제시된 측정 절차에 맞추어, 본 논문에서 구현한 NIC과 기존 방식의 고장 극복용 NIC을 비교 측정하여 Table 5에 제시하였다. 기존 방식은 패킷 전송 시 고장 극복 시간을 단축하기 위해 모든 패킷을 일정시간동안 내장 메모리에 저장하므로 throughput 특성이 현저히 감소하였다. 즉 제안 방식 대비 1/100 수준으로 감소하였으며, 특히 64byte 패킷 전송 경우 1/200 이하로 성능저하를 나타내었다.

Table 5. Test Results for throughput (unit : Kbps)

		Proposed NIC			Conventional NIC		
		Max.	Min.	Ave.	Max.	Min.	Ave.
Packet Size	64bytes	5,979	5,995	5,990	27.4	31.7	29.3
	256bytes	9,676	9,704	9,691	84.5	88.0	85.9
	512bytes	10,810	10,813	10,813	116.4	124.8	120.2
	1,024bytes	11,466	11,468	11,468	144.4	153.0	148.9
	1,518bytes	11,285	11,283	11,285	141.1	198.6	147.3

Ⅳ. 결 론

본 논문은 단일 MAC을 사용하는 새로운 하드웨어 방식 자동 고장 극복 Ethernet (FTE) NIC을 제안하였다. 제안 방식은 Figure 2에 제시한 바와 같이, 두 개의 PHY chip과 단일 MAC 그리고 MII/GMII 신호를 이용하여 port를 선택한다. 단일 MAC과 일반적인 logic gate로 port를 전환하도록 구성하였으므로 상위 어플리케이션에서 사용하는 MAC 주소와 외부 통신을 위한 MAC 주소 매칭을 프로세서를 이용하여 수행하는 기존 고장 극복 NIC과 비교하여 우수한 성능이 나타날 것으로 판단하고, 이를 구현하였으며 실험을 통하여 타당성을 입증하였다.

Table 4에 제시한 바와 같이, 고장 극복 시간은 기존 하드웨어 방식 NIC 장치 대비 동일 또는 우수한 성능을 나타내었다. 또한, 전체 네트워크 시스템의 전송 용량을 결정하는 중요한 특성인 throughput 성능은 Table 5에 제시한 것과 같이 제안 방식은 기존 방식과 비교하여 약 100배 우수한 성능을 나타내었고, 특히 64byte 패킷에서는 200배 이상 높은 성능을 나타내었다. 이는 기존 방식은 내장 co-processor가 전송 패킷 분석시간 소요 이외에도 빠른 고장 극복 시간을 지원하기 위해 전송되는 패킷을 내장 메모리에 일시 저장함으로써, 패킷 전송에 많은 시간이 소요되는데 기인된다.

본 논문은 새로운 Ethernet 이중화 NIC에 관한 설계 개념 제시와 그에 따른 100Mbps NIC 구현 및 실험에 의한 우수한 성능 입증을 다루고 있으나, 제안된 설계개념은 Gbps급 NIC에 확장 적용 가능하다. 그러나 향후 실제 구현 시 Gbps급 고속데이터로 인한 회로상의 노이즈 및 신호 간섭 처리 등의 연구는 추가적으로 필요하다고 판단된다.

References

[1] J.W Shin, and D.S Park, "The Implementation of Fault-Tolerant Dual System Using the Hot-Standby Sparing Technique", *The Journal of the Korean Institute of Communication Sciences*, vol. 29, no. 10A, pp. 1113-1229, Oct, 2004.

[2] 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", *in Proc. 18th IEEE Int. Conf. Performance, Computing, and Communications*, pp. 461-468, Phoenix, USA, Feb. 1999.

[3] S. Song, J. Huang, P. Kappler, R. Freimark, J. Gustin, and T. Kozlik, "Fault-Tolerant Ethernet middleware for IP-Based Process Control Networks", *in Proc. 25th Annual IEEE Int. Conf. on Local Computer Networks (LCN 2000)*, pp. 116-125, Nov. 2000.

[4] H.A Pham, J.M Rhee, S.M Kim, and D.H Lee, "A Novel Approach for Fault-Tolerant Ethernet Implementation", *in Proc. 4th Int. Conf. on Networked Computing, and Advanced Information Management 2008 (NCM'08)*, vol. 1, pp. 58-61, Gyeongju, Korea, Sep. 2008.

[5] S.M Kim, H.A Pham, and J.R Rhee, "Fault Tolerant Ethernet (FTE) Technologies for Mission-critical Military Network Systems {네트워크 기반 첨단 무기체계의 Fault Tolerant Ethernet (FTE) 기술}", *The Journal of The Korean Institute of Communication Sciences*, Vol. 26, no. 3, pp. 69-75 Mar. 2009.

[6] Dong Ho Lee, You-Ze Cho, Hoang-Anh Pham, Jong Myung Rhee, and Yeonseung Ryu, "SAFE: A Scalable Autonomous Fault-tolerant Ethernet Scheme for Large-scale Star Networks", *IEICE TRANSACTIONS on Communications,* vol. E95-B, no. 10, pp. 3158-3167 Oct. 2012.

[7] H.S Kim, Y.C Choi, and W.H Sung, "Gigabit-ethernet based redundant network Performance Analysis", *in Proc. 20th Joint Conference on Communications and Information(JCCI2010)*, Apr. 2010.

[8] GE Fanuc Intelligent Platforms, *PMC677TX*, Retrieved Aug. 26, 2012, from http://www.artisan-scientific.com/66764.htm, or http://www. gefanuc.com

[9] H.A Pham, J.M Rhee, Y.S Ryu, and D.H Lee, "Performance Analysis for a Fault-Tolerant Ethernet Implementation Based on Heartbeat Mechanism", *in Proc. 2011 Spring Conf. The Korea Institute of Information, Electronics, and Communication Technology*, pp. 44-48, Korea, May 2011.

[10] H.A Pham, D.H Lee, and J.M. Rhee, "A Flexible Methodology of Performance Evaluation for Fault-Tolerant Ethernet Implementation Approaches", *in Proc. Int. Conf. Advanced Software Engineering & Its Applications (ASEA 2011)*, Jeju Island, Korea, Dec. 2011.

[11] *eTTCP program*, Retrieved Aug. 26, 2012, from http://www.acq.osd.mil/ttcp/

김 세 목 (Se-Mog Kim)

1998년 2월 부경대학교 전자공학과 졸업

2007~현재 명지대학교 정보통신공학과 석·박사통합과정

<관심분야> Fault Tolerant System, HFC Network

Anh Hoang Pham

2005년 3월 Ho Chi Min 대학교 (베트남 소재) 컴퓨터공학과 졸업

2010년 2월 명지대학교 정보통신공학과 석사

2010년 3월~현재 명지대학교 정보통신공학과 박사과정

<관심분야> Fault Tolerant System, DSP

이 동 호 (Dong-Ho Lee)

1990년 2월 경북대학교 전자공학과 졸업
1992년 2월 경북대학교 전자공학과 석사
2005년 3월~현재 경북대학교 전자공학과 박사과정
1992년 3월~현재 국방과학연구소 책임연구원
<관심분야> Fault Tolerant System, Network Architecture

이 종 명 (Jong Myung Rhee)

1976년 2월 서울대학교 전자공학과 졸업
1978년 2월 서울대학교 전자공학과 석사
1987년 12월 North Carolina State University, ECE Dept 공학박사
1978~1997년 국방과학연구소 책임연구원
1997~1999년 데이콤 연구소 부소장
1999~2005년 하나로텔레콤 CTO (부사장)
2006년~현재 명지대학교 정보통신공학과 교수
<관심분야> Military Communication, Fault Tolerant System, Ad-hoc, Data Link, Convergence, Smart Grid Communications