고신뢰성 발사통제시스템을 위한 고장허용 통신 미들웨어 설계 및 구현

Design and Implementation of Fault-tolerant Communication Middleware for a High-reliable Launch Control System

송대기*, 장부철*, 이철훈**
국방과학연구소*, 충남대학교**

Dae-Ki Song(dksong@add.re.kr)*, Bu-Cheol Jang(bcjang@add.re.kr)*,
Cheol-Hoon Lee(clee@cnu.ac.kr)**

요약

발사통제시스템(Launch Control System)은 유도무기체계에서 유도탄의 발사와 관련된 일련의 절차들을 제어하는 시스템이다. 이 시스템은 지정된 시간 안에 목표물에 대한 교전계획을 생성하고 이 정보를 탄에 장입하며 탄의 발사 제어를 수행해야 한다. 이 때문에 시스템의 하드웨어, 소프트웨어 고장뿐만 아니라 정해진 수행 시간의 초과 때문에도 임무가 실패될 수 있다. 본 논문에서 제안한 것은 발사통제시스템과 같은 내장형 실시간 시스템을 위한 고장허용 이더넷으로 별도의 하드웨어나 소프트웨어 없이 기존 상용의 이더넷 디바이스를 이중화하여 네트워크에 고장허용 기능을 제공하는 방법이다. 이를 위해 시스템의 각 구성 노드들을 이중의 네트워크 채널로 중복 시키고, 네트워크 트래픽에 대한 고장탐지 및 복구를 위해 통신 미들웨어를 설계하고 개발하였다. 이중채널 기반의 통신 미들웨어를 통해 처리 시간이 중요한 시스템에 네트워크 고장으로 인한 시스템 중지를 방지하고 노드에 통신 손실이 발생하지 않도록 하였다.

■ 중심어 : | 발사통제 시스템 | 임베디드 시스템 | 고장허용 통신 미들웨어 | 이중 채널 통신 |

Abstract

Launch control system controls the sequence for launching missile in weapon systems. This system have to generate the engagement plan, input information and launch the missile in timeliness requirement. Such a system may fail to operate correctly either due to errors in hardware and software or due to violation of timing constraints. We presented fault-tolerant ethernet for embedded real-time system like launch control system. This approach is designed to handle network faults using dual commercial-off-the-shelf(COTS) network devices. To support fault-tolerant ethernet each node is composed dual channel ethernet and designed the communication middleware for network fault detect and recovery. Especially for time-critical system, the middleware is being developed to achieve that no single point of network failure shall take down or cause loss of communication to network nodes.

■ keyword : | Launch Contorl System | Embedded System | Fault-tolerant Communication Middleware | Dual-channel Communication |

접수번호 : #080624-002
접수일자 : 2008년 06월 24일
심사완료일 : 2008년 07월 07일
교신저자 : 송대기, e-mail : dksong@add.re.kr

I. 서 론

오늘날의 컴퓨터는 초창기 컴퓨터 시스템처럼 단순 과학계산에만 사용되는 것이 아니라 은행 업무에서부터 시작하여, 국방 분야, 증권거래, 항공기 관제 업무, 기상관측 그리고 개인용도에 이르기까지 사회 전반에 걸친 다양한 분야에서 사용되고 있다. 특히 국방 분야, 핵 발전, 은행업무나 증권거래와 같이 중요한 업무에서의 컴퓨터 고장은 막대한 인명 및 재산 피해를 초래할 수 있다. 이 때문에 여러 회사들은 컴퓨터에서 발생할 수 있는 고장에 대처하기 위해 다양한 방안들을 개발하였고 이를 실제 적용한 컴퓨터를 생산하여 필요한 분야에 활용하고 있다[1]. 일반적으로 고 신뢰도(High Reliability)에 대한 요구를 만족시키기 위한 시스템을 고장허용(Fault-Tolerant) 컴퓨터라고 부른다. 고장허용 컴퓨터 시스템이라 함은 시스템의 구성 요소 중의 하나에 고장이 발생하더라도 주어진 임무를 계속 수행하도록 설계한 시스템을 의미한다. 고장허용으로 분류되는 시스템들은 시스템내의 모든 단일점고장(SPOF: Single Point Of Failure)에 대하여 감내할 수 있는 시스템으로 규정짓고 있다[2][3].

유도무기를 위한 발사통제시스템(Launch Control System)은 일종의 내장형 실시간 시스템으로써 유도탄을 발사하기 위하여 실시간으로 탄 상태를 모니터링하면서, 정해진 시간에 교전계획 및 비행정보를 생성하여 탄에 입력하고 전지, 구동장치 및 부스터와 같은 발사에 필요한 다양한 착화 신호들을 제어해야한다. 일반적으로 발사통제시스템은 세부기능에 따라 몇 개의 장비들로 구성되고 각 장비들은 이더넷, 시리얼통신 및 MIL-STD-1553B[4] 등과 같은 여러 방식으로 통신망을 구성하여 데이터를 주고받는다. 발사통제시스템은 지정된 목표물에 대하여 정해진 시간 내에 유도탄을 발사해야 하는 시간 결정적(Time-critical) 시스템으로 하드웨어 및 소프트웨어의 고장뿐만 아니라 시간 제한사항을 만족하지 못하는 것만으로도 시스템의 고장을 유발할 수 있다[5].

시스템에는 여러 종류의 단일고장점이 존재할 수 있지만 네트워크측면에서 보았을 때 네트워크 인터페이스 카드(NIC), 허브(HUB), 네트워크 스위치(Network Switch), 연결 케이블 등의 네트워크 디바이스들이 단일고장점이 될 수 있다. 이 때문에 내장형 실시간 시스템을 위한 다양한 하드웨어 및 고장허용 통신 방식이 구현되어 있지만 이들의 대부분은 특화된 하드웨어나 프로토콜을 필요로 하는 독자적인 방식을 취하고 있어 범용 환경에 적용시키기가 쉽지 않다[6-8]. 특히 발사통제와 같은 시간 결정적 시스템에는 네트워크 오류 및 고장으로 인한 전송 지연 대하여 치명적이기 때문에[6] 노드 간 신뢰성과 실시간성을 보장하는 통신 기능이 필수적이다.

본 논문에서는 발사통제 시스템과 같은 내장형 실시간 시스템의 구성 노드 간 통신 채널의 신뢰성과 실시간성을 보장하기 위하여 이중 채널 기반의 고장허용 통신 미들웨어(이하 FTCM : Fault-Tolerant Communication Middleware)를 제안하였다. FTCM은 개방형 구조의 어플리케이션 미들웨어로 별도의 하드웨어나 소프트웨어 없이 상용(COTS)의 이더넷 하드웨어(NIC, Switch HUB 등)와 소프트웨어(데이터링크 계층 및 드라이버)만으로도 고장허용 기능을 제공할 수 있다. 또한 응용 단계에서 미들웨어 소프트웨어를 구현함으로써 다양한 운영체제로의 포팅이 용이하게 하였고, 시스템 특성에 맞게 다양한 고장 허용 기법이 적용가능하다. FTCM은 시스템 네트워크의 어느 한 지점에 고장이 발생하더라도 이와 독립적으로 주어진 임무를 요구된 시간 안에 수행하는 것을 첫 번째 목표로 하고 있다. 본문에서는 발사통제시스템의 특징과 FTCM의 개념 및 구조, 고장허용 기능에 대하여 설명하고 실험에서 측정한 결과 분석 내용을 기술한다.

II. 관련연구

1. 고장허용 이더넷

고장허용 이더넷은 주로 공정 제어 산업을 위하여 설계되고 적용되어 왔으며[7], 대부분이 특화된 하드웨어를 사용하고 이를 위한 독자적인 소프트웨어를 사용하고 있다. 하지만 최근에 상용 네트워크 장비 및 표준들

이, 특히 대중적이고 저가인 이더넷, 고장 허용 네트워크를 개발하기 위한 기반이 되어가고 있고 정보 기술(IT : Information Technology) 산업의 PC 및 네트워크 장비 업체들도 중요 업무(Mission Critical) 시스템을 위한 이더넷 제품들을 출시하고 있다. 그러나 이것들은 서버-클라이언트 모델을 위하여 설계되었기 때문에 다양한 하드웨어 플랫폼을 사용하고 주로 일대일 통신을 하는 내장형 실시간 시스템에는 이러한 제품들이 적당하지 않다. 그동안 여러 기업에서 다양한 고장허용 이더넷 방법을 연구하여 왔다. 이중 3COM, Intel, GE와 같은 업체들은 이중 링크를 통해 장애극복(Fail over)을 지원하는 EIC(Ethernet Intelligent Controller)들을 개발하였다. 이 카드들은 하드웨어적인 방법으로 고장을 탐지하고 EIC 서로 간에 통신 링크를 교환하기 때문에 비교적 쉽게 고장허용 네트워크를 구현하고 빠른 고장 복구를 보장할 수 있다. GE의 Ramix675 EIC의 경우 링크 절체시 연결 특성이 유지되고 50ms 이하의 고속 고장 감지 및 경로 절체 기능을 지원한다. 하지만 이 EIC들은 상당히 고가의 제품들이고 다양한 플랫폼(하드웨어, 운영체제)을 지원하지 않기 때문에 내장형 시스템 개발 시 여러 제약조건이 발생한다.

2. 하니웰 고장허용 이더넷

하니웰((Honeywell) 사에서 제안 한 고장허용 이더넷은 상용의 이더넷 디바이스를 이중화하여 고장허용 기능을 제공하는 것으로, 네트워크 데이터 링크 계층과 트랜스포트 계층 사이에 별도의 미들웨어를 두어 고장탐지 및 복구 기능을 제공한다. [그림 2]에서 보면 미들웨어는 NIC Switch와 FT Manager의 두 부분으로 구성되어 있다. NIC Switch는 데이터 링크 계층에서 두 개의 NIC 드라이버 사이의 채널 교체를 수행하고, FT Manager는 고장 탐지 및 복구 알고리즘을 수행한다. FT Manager는 이중 채널에 대하여 주기적으로 핫비트(Heartbeat) 메시지를 전송하여 이에 대한 응답 확인으로 채널의 고장 유무를 판단한다. 고장이 탐지된 경우 이를 NIC Switch로 통보하여 데이터 통신 채널을 교환하여 네트워크 고장을 복구한다. 하니웰 고장허용 이더넷은 Hot-Standby 이중채널 방식으로 사용한다. 하지만 이 방법은 고장 탐지에서 복구 시까지 시간 지연이 발생하기 때문에 그 값만큼 실시간 시스템의 처리 수행 시간이 증가할 수 있고, 이 때문에 시스템의 실시간성이 깨질 수 있다. 하니웰 고장허용 이더넷의 고장탐지 시간은 아래의 식(1)과 같다.

$$\textit{Failure detection time} = (MaxLossAllowed + 1) \times T_p + T_{skew} \quad \text{(Seconds)} \quad (1)$$

[그림 1]은 Node i와 Node j 간 이중 채널 사이의 핫비트 메시지 전송을 도시한 것으로, T_p는 핫비트 메시지의 전송주기이며, T_{skew}는 두 채널 사이의 핫비트 메시지 도착 시간차이, MaxLossAllowed 값은 고장을 탐지되기 위한 핫비트 메시지 손실 개수를 의미한다. 만약 MaxLossAllowed 값이 1일 경우 고장탐지 시간은 $2T_p + T_{skew}$ (Seconds) 가 된다.

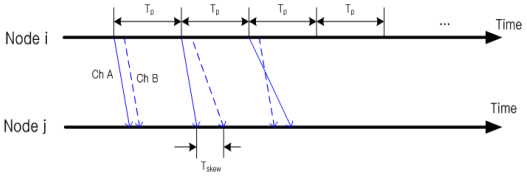

그림 1. 고장탐지를 위한 핫비트 메시지 전송

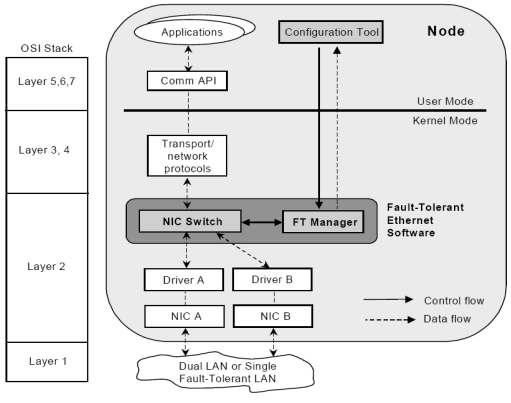

그림 2. 하니웰의 고장허용 이더넷 구조

고장탐지 시간은 허용된 핫비트 메시지 손실 개수와

핫비트 메시지 전송주기, 이중 채널 사이의 핫비트 메시지 수신 시간 차이에 따라 결정된다. 고장탐지 시간이 길게 되면 두 통신 노드 상에 메시지의 손실이 발생할 수 있고 이로 인하여 시스템의 오동작 및 고장이 유발될 수 있다. 고장탐지 시간을 줄이기 위하여 핫비트 손실 메시지 개수를 조정하거나 핫비트 메시지 전송 주기를 짧게 하면 되지만 전자의 경우는 채널상의 일시적인 네트워크 오류에 민감하게 채널 교환이 발생할 수 있고, 후자의 경우는 채널 상에 네트워크 부하를 키우는 문제가 있다. 이와 같은 고장허용 이더넷에서 고장복구에 소요되는 지연 시간은 시간 결정적 시스템에서 실시간성을 깨는 요인이 될 수 있기 때문에 이를 발사통제시스템에 적용하는데 문제가 있다.

III. 고신뢰성 발사통제시스템

유도무기를 위한 발사통제시스템은 유도탄의 발사와 관련된 일련을 절차들을 제어하는 시스템으로 사격 임무를 위하여 탐지기능, 표적 획득 기능, 추적기능, 발사기능, 유도기능이 필요하며, 무기체계 특성에 따라서 발사통제시스템에 그 기능이 추가되거나 삭제된다. 발사통제시스템은 언제 발생할지 모르는 적 목표물에 대하여 항상 대비해야 하고, 표적 정보 획득 후 즉각적으로 교전 계획 생성과 유도탄 발사 절차를 수행할 수 있어야 한다. 만약 시스템 내 한 모듈에 고장이 발생하면 장착된 유도무기를 운용할 수 없게 되며, 임무 수행, 장비 및 인력 등에 심각한 위험을 주게 된다. 발사통제시스템의 구성 노드들은 VMEBus를 사용하는 내장형시스템으로 온도, 습도 등의 군용 환경 규격을 만족하도록 설계된 컴퓨터 및 입출력 보드들로 구성된다. 발사통제시스템의 보드들은 군용 장비 특성 상 신뢰성이 보장되어야하며 개발 후 장기간에 걸쳐 운용되기 때문에 부품과 모듈 등의 수명 또한 고려하여 시스템 하드웨어를 설계하여야 한다. 노드의 전산기 모듈은 단일보드컴퓨터(Single Board Computer)로 군용 시스템의 환경조건에 맞도록 러기드(Rugged)되어 있으며 신뢰성이 뛰어나다. 하지만 범용 컴퓨터 시스템과 비교하여 상대적으로 성능이 낮고, 다양한 입출력 장치를 사용하지 못하는 단점을 갖고 있다. 현재는 표준 VME 버스 기반의 상용 전산기 보드와 상용 실시간 운영체제를 사용하고 있지만 과거에는 별도의 하드웨어 플랫폼위에 독자적인 운영체제 및 소프트웨어를 사용하였다. 구성 장비간 통신도 직렬통신 방식의MIL-STD-1553B[4] 통신 프로토콜이 주로 사용되었는데 1970년대에 군용으로 개발되어 지금까지 무기체계 또는 항공기 체계에서 널리 사용되고 있다. MIL-STD-1553B는 Half-Duplex 직렬통신 방식으로 최대 전송 속도가 1 Mbps로 저속이며 한 번에 전송하거나 받을 수 있는 데이터의 길이가 최대 32 Words(1Word=16bits)로 제한된다는 단점이 있다. 현재는 유도탄과 인터페이스 하는 장비를 제외하고는 이더넷을 이용하여 통신을 수행하고 있다.

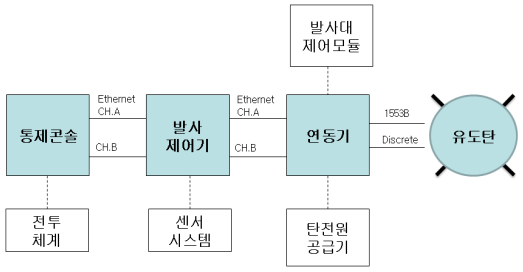

그림 3. 발사통제시스템 구성

발사통제시스템은 [그림 3]과 같이 기능에 따라서 몇 개의 장비들로 구성되며, 전투체계(Combat System), 센서 시스템(Sensor Systems), 탄전원공급기(Missile Power Supplier), 발사대제어모듈(Launcher Control Module) 등의 장비들과 직간접적으로 연동되어 임무를 수행한다. 무장 종류와 발사 플랫폼에 따라 장비 구성 및 기능이 다양하지만 일반적으로 발사제어를 수행하는 통제콘솔, 센서 시스템들과 연동하는 제어기, 탄과 인터페이스 하는 연동기, 탄에 전원을 공급하는 탄전원 공급기 등의 장비들로 구성된다. 그림에서 통제콘솔은 탄 제어 명령과 교전계획을 생성하고 탄 정보 및 환경정보를 모니터하며 실제적인 발사절차를 수행한다. 이 장비는 탄 정보 및 발사대 상태 정보를 일정 주기로 수신하며 모니터링하고 발사에 필요한 제어 신호를 절차

및 시간에 맞게 탄에 전송한다. 발사 제어기는 센서 시스템으로부터의 정보와 발사대 상태 정보를 수신하여 콘솔 및 연동기로 제공하는 역할을 수행한다. 연동기는 유도탄과 직접 인터페이스 하는 장비로 콘솔에서 전송된 제어 명령을 받아서 유도탄 발사에 필요한 이산 제어 신호를 생성하고 유도탄 내부의 유도조종 컴퓨터(Guidance Control Computer)와 통신을 수행하여 탄 내부 상태를 읽어 상위 장비로 전송하는 역할을 수행한다. [표 1]은 각 노드의 태스크들을 정리한 것이다.

표 1. 발사통제시스템 노드 별 태스크 할당

노드명	태스크명	설명	태스크주기
통제콘솔	MissileMonitor	유도탄 상태 모니터링	50ms
	LauncherMonitor	발사대 상태 모니터링	200ms
	LaunchControl	발사제어 (제어 명령 생성)	20ms
	InterfaceGUI	영상부와 인터페이스	비주기
	InterfaceC2	전투체계와 인터페이스	1sec
발사제어기	LauncherControl	발사대 제어	비주기
	InterfaceSensor1	센서1과 인터페이스	100ms
	InterfaceSensor2	센서2와 인터페이스	10ms
연동기	MissileControl	유도탄 제어	비주기
	1553BCOM	탄 통신 (MIL-STD-1553B)	200ms
	DiscreteControl	이산신호 출력 제어	비주기
	DiscreteMonitor	이산신호 모니터링	50ms
	InterfaceLauncher	발사대 인터페이스	200ms

발사통제시스템에서 연동기와 유도탄과의 통신은 MIL-STD-1553B의 이중 채널을 통한 자체 고장 복구가 지원되지만 그 외 장비 간 이더넷은 네트워크 고장에 대한 대비가 되어 있지 않기 때문에 추가적인 하드웨어 및 소프트웨어가 필요하다. 기존 발사통제시스템의 경우 네트워크 인터페이스 카드, 허브, 통신 케이블을 이중으로 구성하고 이를 Hot-Standby[8] 형태로 운용하였다. 그러나 2장에서도 언급했듯이 Hot-Standby 형태는 고장탐지에서 복구까지 시간지연이 불가피하기 때문에 즉각적인 반응을 요구하는 전투상황 또는 발사절차 진행 하에서 이로 인해 군의 임무 수행이 실패로 끝날 수 있다. 다음 장에서는 본 논문에서 제안한 고장허용 이더넷의 특징과 구조에 대하여 기술하겠다.

IV. 고장허용 통신 미들웨어 구조

1. 이중 채널 구조

발사통제시스템은 유도탄 통제콘솔, 발사제어기, 연동기 등의 네트워크 노드들로 구성되며, 유도탄 및 발사대를 실시간으로 제어하고 모니터링하기 위해 이더넷을 통해 데이터를 주고받는다. 이러한 제어용 네트워크는 어느 한 지점의 결함으로 인하여 데이터가 손실되거나 통신 불능 상태가 될 수 있으며, 이로 인해 시스템 전체에 고장이 발생할 수 있다. 때문에 네트워크 인터페이스 카드, 링크 및 허브 등을 이중화하여 시스템을 구성한다. 논문에서 제안한 기법은 기존의 Hot-Standby 이중채널과 달리 데이터 통신을 위해 두 채널 모두를 동시에 사용하는 Hot-Spare 이중채널 기법을 사용하였다.

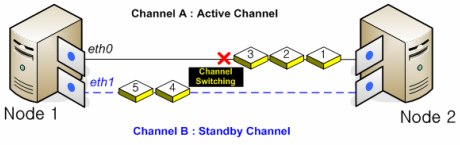

그림 4. Hot-Standby 이중채널 통신

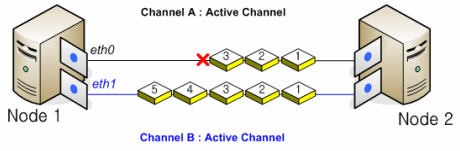

그림 5. Hot-Spare 이중채널 통신

[그림 4]와 같은 기존 방식은 고장 발생 시 탐지 및 채널 교환 때문에 두 노드 사이의 통신에 시간 지연이 발생하고 이 때문에 실시간성이 깨질 수 있다는 문제가 있다. [그림 5]의 Hot-Spare 이중채널 방식은 데이터 전송 시 두 채널을 통해 동시에 이루어지기 때문에 어느 한 채널에 고장이 발생하더라도 나머지 채널에 의해 데이터가 전송되며 또한 복구를 위한 고장 탐지 및 채널 교환이 필요하지 않고 두 노드간의 실시간 통신을

보장할 수 있다. 다만 데이터 전송을 위해 두 채널 모두를 사용하기 때문에 전송 및 수신 노드에 네트워크 부하가 증가하고 각 노드의 어플리케이션에서 이중 채널의 동일 메시지를 처리해야 하는 문제가 발생한다. 때문에 본 논문에서는 이를 위한 어플리케이션 레벨의 통신 미들웨어를 설계 및 제안한다.

2. 고장허용 통신 미들웨어(FTCM) 구조

미들웨어는 복잡한 이기종 환경에서 어플리케이션과 운영환경 간에 원만한 통신을 이룰 수 있게 해주는 소프트웨어로 다양한 하드웨어, 네트워크 프로토콜, 응용 프로그램, 근거리통신망 환경, PC 환경 및 운영체제의 차이를 메워주는 역할을 한다. 고장허용 통신 미들웨어(이하 FTCM)는 응용 프로그램과 이중 이더넷 디바이스 간의 인터페이스 역할을 하며 사용자에게 고장허용 기능에 대한 투명성을 제공한다. 응용 프로그램 개발자는 FTCM을 이용하여 손쉽게 고장허용 어플리케이션을 개발할 수 있도록 도와준다. 미들웨어 소프트웨어는 [그림 6]과 같이 OSI 계층 구조상 전송계층(Transport Layer)의 상위에 구현되었다. 어플리케이션 계층에서 미들웨어를 설계함으로써 여러 운영체제로의 포팅과 다양한 고장허용 기법 적용을 용이하게 하였다.

본 논문에서 제안한 고장허용 이더넷은 일반적인 분산 컴퓨팅 환경과는 다른 내장형 시스템이라는 제한적인 환경(하드웨어,운영체제,이더넷 라이브러리 등)을 목적으로 개발되었다. 때문에 다양한 하드웨어, 소프트웨어 및 통신 프로토콜을 사용하는데 제약을 받으며, 발사통제라는 특수목적에 따라 실시간 통신을 수행해야 한다는 점과 각 노드 간 일대일 통신을 사용한다는 점이 고려되었다. FTCM에서 제공하는 API를 사용함으로써 사용자는 이중 채널을 통한 메시지 송수신 및 고장 탐지, 복구 기능 등을 서비스 받는다. FTCM은 발사통제시스템에서 운영체제로 사용하고 있는 QNX4.25를 기반으로 설계되었고, 운영체제에서 제공하는 IPC와 API를 사용하여 개발되었다. 미들웨어의 API는 기존 운영체제에서 제공하는 API와 유사하도록 하여 소프트웨어 이식성을 높였다.

고장허용 미들웨어의 구조는 아래 그림과 같이 고장 탐지 관리자(Fault Detect Manager), 고장허용 통신 관리자(FT Communication Manager)와 미들웨어 자원 관리자(Middleware Resource Manager)로 구성되어 있다. 고장 탐지 관리자는 통신 채널에서 메시지를 송수신하면서 발생하는 통신 오류를 감지하거나 하트비트(Heartbeat) 메시지를 사용하여 네트워크 링크 상태 및 각 노드들의 상태를 관리한다. 고장 탐지 관리자는 시스템을 구성하는 노드들의 상태를 NST(Node Status Table)로 관리하는데 이 정보는 고장허용 통신 관리자에게 제공되어 미들웨어 계층에서 메시지를 송수신하는데 사용된다. 고장허용 통신 관리자는 실제적인 메시지 송수신 업무를 처리하는데, 사용자가 FTCM의 API를 호출할 경우 고장허용 통신 관리자는 이를 이중 채널을 이용한 메시지 송수신 함수로 변환하여 처리한다. 미들웨어 자원 관리자는 메시지를 위한 캐쉬 영역을 관리하는데 이는 메시지를 재전송하고, 동일 메시지 검사를 위한 임시 메모리 저장소로 사용된다. FTCM의 각 관리자 프로그램들은 쓰레드(thread)로 생성되어 자신의 기능들을 수행한다. 미들웨어는 메시지를 송신하기 위한 쓰레드, 각 채널에서 메시지를 수신하기 위한 쓰레드, 자원 관리를 위한 쓰레드, 고장 탐지를 위한 쓰레드로 구성된다.

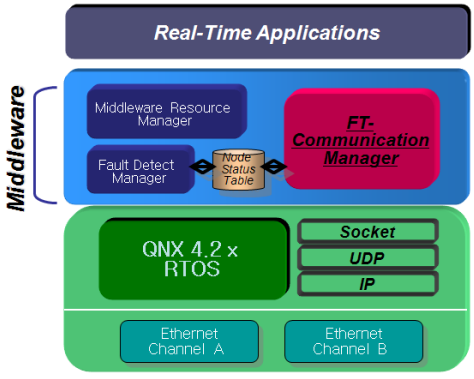

그림 6. FTCM 구조

응용프로그램이 메시지를 전송하는 경우 [그림 7]의 3, 4와 같이 미들웨어에서 자동으로 메시지를 이중화하여 두 채널로 전송하고, 수신하는 경우에는 [그림 7]의 6과 같이 두 채널 중 먼저 도착한 메시지를 어플리케이

션에 전달하고 후에 도착한 메시지는 같은 메시지임을 확인 후 제거한다. 고장허용 통신 관리자는 UDP 계층에서 각 메시지에 체크썸(Checksum) 필드를 추가하여, 어플리케이션과 독립적으로 미들웨어 단계에서 메시지의 내용이 유효한지 검사한다. 또한 신뢰성 있는 통신을 위하여 메시지 전송 시에 ACK(Positive Acknowledgement) 메시지와 NACK(Negative Acknowledgement) 메시지를 사용하여 전송 결과를 확인하도록 하였다. 메시지 수신이 완료된 경우 ACK를 전송하여 송신자에게 정상적으로 수신했음을 알리며, 만약 메시지 체크썸에 이상이 있는 경우 NACK를 사용하여 메시지 내용에 오류가 있음을 통보한다. 송신자의 경우 메시지 전송 후 해당 이벤트에 대한 타임아웃(Timeout)을 설정하여, 이 기간 안에 ACK를 수신하지 못하면 동일 메시지를 재전송하도록 하였다. 이를 통해 사용자는 신뢰성이 보장된 통신 기능을 갖게 된다.

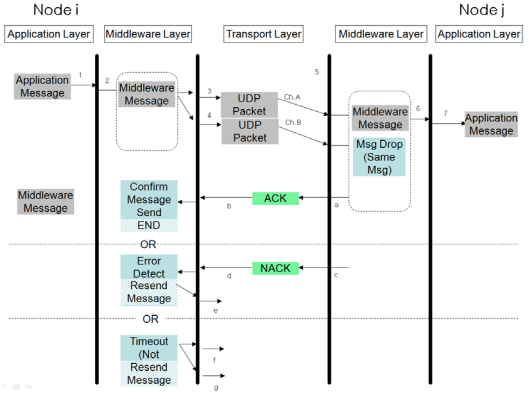

그림 7. 이중채널을 통한 메시지 송수신 절차

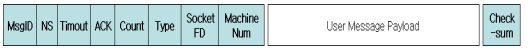

그림 8. 미들웨어 메시지 필드

미들웨어는 내부적으로 메시지를 핸들링하기위하여 사용자 메시지에 [그림 8]과 같은 헤더를 추가한다. 고장허용 통신 관리자는 각 메시지를 식별하기 위하여 MsgID(메시지ID), NS(메시지 순차번호), MachineNum(노드ID) 정보를 이용하고, 메시지의 내용이 유효한지 검사하기 위해 체크썸 워드를 사용한다. 수신단에서 메시지 유효검사 결과가 비정상일 경우 NACK메시지를 전송하여 재전송을 요구하며, 이는 [그림 7]의 c, d, e 절차에 해당한다. Timeout은 메시지 전송 후 수신자로부터 ACK 메시지를 수신할 때까지 증가하는 시간 값으로 이 값이 허용범위를 초과할 경우 통신 관리자는 해당 메시지를 재전송하게 된다. [그림 7]의 e, f에 해당한다. 실시간 응용 소프트웨어는 기존 운영체제에서 제공하는 API 대신 미들웨어의 API를 사용함으로써 하위 네트워크 계층에 대하여 논리적으로 투명성을 갖게 되며 적은 비용으로 쉽게 고장 허용 기능을 구현할 수 있다. 또한 기존 개발 완료 된 소프트웨어에 대하여 통신 API만을 교체하여 고장허용 이더넷 능력을 쉽게 이식할 수 있다.

V. 실험 결과

미들웨어 구현 및 성능 측정을 위하여 [그림 9][그림 10]과 같이 PC를 기반으로 1차 환경을 구성하였다. 1차 환경의 PC에는 발사통제 시스템에서 사용하는 실시간 운영체제인 QNX4.25를 설치하였으며 모든 노드에는 두 장의 NIC를 장착하여 각각 채널A 허브, 채널B 허브에 연결시켜 놓았다. 또한 허브에 네트워크 분석용 컴퓨터를 연결하여 각 채널별 통신 성능 및 링크 절체시의 UDP 패킷 변화를 분석하였다.

1차 실험의 경우 미들웨어 소프트웨어 개발하고 기능 및 성능 측정을 위한 환경으로 4대의 PC로 구성되어 있다. 노드들은 일대일 통신을 수행하며 이중 어느 한 노드의 통신 케이블을 허브에서 제거하여 네트워크 고장을 발생시켰다. 실험에 사용된 응용 프로그램은 아래 [그림 10]과 같이 노드2와 노드3이 각각 20ms 주기로 노드1에 메시지를 전송하고, 노드1은 수신한 데이터를 노드4에 전달한다. 이 실험을 통하여 이중 채널 중에서 한 채널의 고장이 시스템에 미치는 영향을 분석할 수 있는데, 실험 결과 네트워크 채널 고장에 영향을 받지 않고 종단(노드4)까지 데이터가 전송되어 처리됨을 확인할 수 있었다. 노드들의 통신 채널 절체 순서는 (1)노드2 Ch.A 분리 (2)노드2 Ch.A 연결 (3)노드2 Ch.A 분리 (4)노드2 Ch.B 분리로 진행되었다.

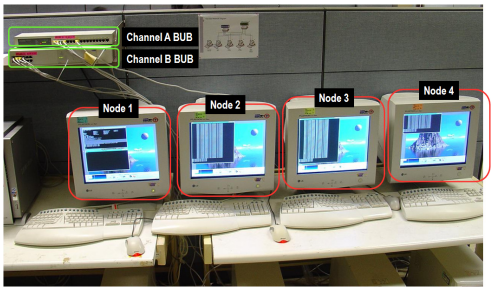

그림 9. 1차 실험을 위한 PC 기반의 실험환경

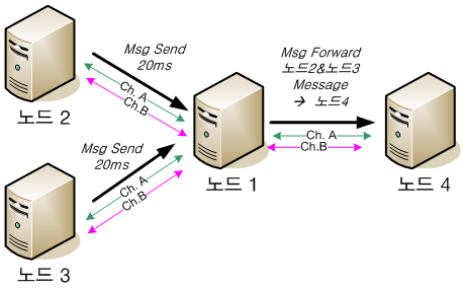

그림 10. 1차 실험의 응용 프로그램

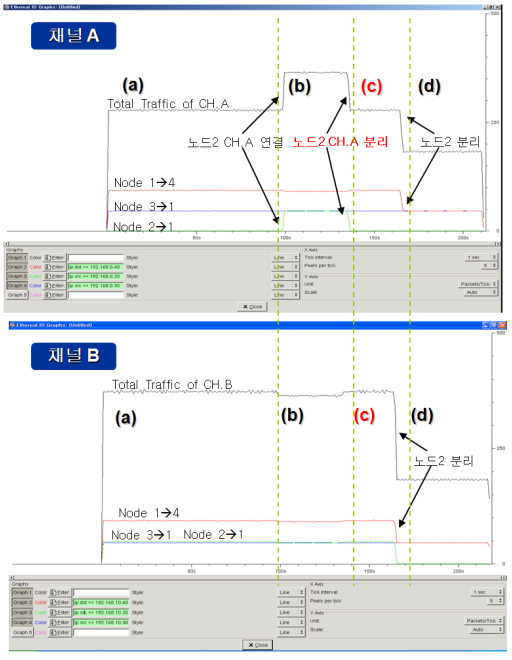

그림 11. 채널A,B의 UDP 패킷 입출력 그래프

[그림 11]에서 검정 선은 총 트래픽 량으로 노드1, 2, 3, 4에서 송/수신 되는 모든 패킷 입출력 크기를 나타내고, 붉은 선은 노드1에서 노드4로 전송되는 트래픽 량을 나타낸다. 녹색선과 파란 선은 각각 노드2와 노드3에서 노드1로 송신되는 트래픽 량을 나타낸다. 그림에서 (a),(b),(c)단계는 두 채널 중 한 채널만 제거했을 경우로, 이 단계의 붉은 선을 통해 노드4로 전달되는 트래픽 량의 변화가 없음을 볼 수 있다. 다만 노드2의 채널 A, B가 모두 절체 되는 경우에는 트랙픽 량이 그림의 (d)단계와 같이 감소한다. 이를 통해 미들웨어의 이중 채널 통신 기능을 확인하였다.

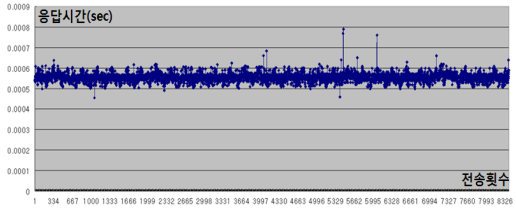

그림 12. 메시지 응답 지연 시간 측정

[그림 12]는 응용프로그램에서 메시지를 전송한 후 그에 대한 응답까지의 시간을 측정한 것으로 약 500~600us의 지연 시간을 보였다. 이를 통해 메시지가 상태 노드의 응용프로그램까지 전송되는 시간을 확인할 수 있다. 또한 가로축은 각 메시지의 순차번호로 통신 시 메시지 손실이 없었음을 확인할 수 있다.

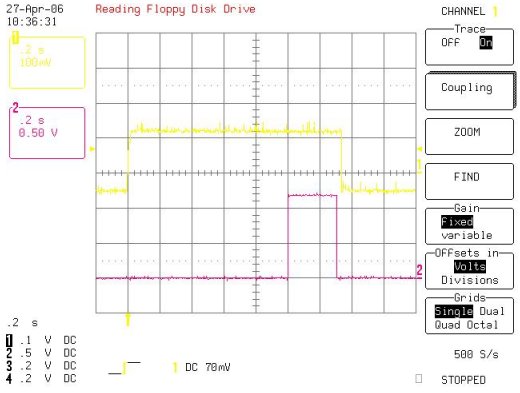

그림 13. 통신 채널 절체시의 탄 제어 신호 측정
(1) 추진장치 장전 신호 (2) 발사 신호

2차 실험의 경우 실제 배치될 발사통제시스템에 미들웨어를 포팅하여 구성하고 그 성능 및 기능을 평가하였다. 다만 유도탄을 연결하여 성능 시험을 할 수 없기 때문에 별도의 모의기를 사용하여 유도탄 및 발사대 상태를 모의토록 하였다. [그림 13]에 보인 실험 결과는 탄연동장비에서 출력되는 제어 신호를 직접 측정한 것으로, 유도탄 발사제어 과정 중간에 장비 간 통신 채널인 광케이블을 장비에서 제거한 후 경우이다. 1번(황색)은 추진장치 장전 신호이고 2번(적색)은 발사 신호로써, 케이블이 제거된 상황 하에서도 1번의 장전 신호 후 정확히 1초 후에 발사 신호가 출력됨을 확인할 수 있다. 이 단계 이외에도 발사절차 및 운용 시에 링크 절체, 노드 제거 시험을 통하여 고장허용 미들웨어가 실제 장비에서 원활히 동작함을 확인하였다.

VI. 결 론

본 논문에서 제안한 고장허용 이더넷 기법은 발사통제시스템과 같은 시간 결정적 시스템을 위한 것으로 이중채널을 기반으로 실시간성을 보장하면서 통신의 신뢰성을 보장할 수 있다. 별도의 하드웨어 없이 NIC, HUB, 통신케이블을 중복하여 이중 채널을 구성하고 이를 위한 API를 제공하기 위해 미들웨어를 구현하였다. 기존 기법과 달리 두 채널을 동시에 사용토록 하여 어느 한 시점에 고장이 발생하여도 나머지 채널에 의해 통신이 수행될 수 있고, 미들웨어를 사용함으로써 사용자에게 이중 채널 사용에 대한 투명성과 신뢰성 있는 통신 서비스를 제공한다. 이 기법은 분산컴퓨팅 환경보다는 내장형 실시간 시스템과 같이 각 노드 간 일대일 통신을 하는 시스템을 목적으로 개발되었다. 발사통제시스템에 FTCM을 적용함으로써 네트워크 고장으로 인해 처리 작업이 중단되고 실시간성이 깨지는 것을 방지할 수 있었다. 추후 진행될 연구에서는 성능 개선 및 안정화에 대한 부분과 다양한 내장형 실시간 시스템에 적용하기 위한 범용화 작업이 필요하다.

참 고 문 헌

[1] 김문회, “결함 허용 시스템의 설계 고려사항 및 동향”, 정보과학회지, 제11권, 제3호, pp.7-16, 1993(6).

[2] S. Hariri, A. Choudhary, and B. Sarikaya, "Architecture support for designing fault tolerant open distributed system," IEEE computer, pp.50-62, 1992.

[3] C. Lee, "TOPIX: theory of operation," SEC Tech. Prt., S40-OSTPX-PC, 1992.

[4] "MIL-Std-1553B : Digital Time Division Command/Response Multiplex Data Bus," FSC, 1978.

[5] F. Jahanian, "Fault-tolerance in embedded real-time systems," in Hardware and Software Architecture for Fault Tolerance, (eds. M. Banatre and P.A. Lee), pp.237-249, 1994.

[6] S. Song, J. Huang, P. Kappler, R. Freimark, and T. Kozlik, "Fault-Tolerant Ethernet Middleware for IP Based Process Control Networks," Proc. 25th Annual IEEE Conference on Local Computer Networks, Tampa, Florida, USA, pp.116-125, 2000.

[7] 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," Proc. IEEE International Performance, Computing, and Communications Conference, Phoenix/ Scottsdale, Arizona, USA, pp.461-478, 1999.

[8] J. Pankaj, Fault Tolerance in Distributed Systems, Englewood Cliffs, NJ: Prentice-Hall, 1994.

저 자 소 개

송 대 기(Dae-Ki Song) 정회원

- 1997년 2월 : 충남대학교 컴퓨터공학과(공학사)
- 1999년 2월 : 충남대학교 컴퓨터공학과(공학석사)
- 2002년 1월 ~ 현재 : 국방과학연구소 연구원

<관심분야> : 내장형 실시간 시스템, 고장허용 컴퓨팅

장 부 철(Bu-Cheol Jang) 정회원

- 2000년 2월 : 경북대학교 전자공학과(공학사)
- 2002년 2월 : 경북대학교 전자공학과(공학석사)
- 2002년 1월 ~ 현재 : 국방과학연구소 연구원

<관심분야> : 내장형 실시간 시스템, 네트워크

이 철 훈(Cheol-Hoon Lee) 정회원

- 1983년 2월 : 서울대학교 전자공학과(공학사)
- 1988년 2월 : 한국과학기술원 전기및전자공학과(공학석사)
- 1992년 2월 : 한국과학기술원 전기및전자공학과(공학박사)
- 1983년 3월 ~ 1986년 2월 : 삼성전자 컴퓨터사업부 연구원
- 1992년 3월 ~ 1994년 2월 : 삼성전자 컴퓨터사업부 선임연구원
- 1994년 2월 ~ 1995년 2월 : Univ. of Michigan 객원연구원
- 1995년 2월 ~ 현재 : 충남대학교 컴퓨터공학과 교수
- 2004년 2월 ~ 2005년 2월 : Univ. of Michigan 초빙연구원

<관심분야> : 실시간시스템, 운영체제, 고장허용 컴퓨팅