

주기적 지연 데이터를 위한 알고리즘 설계 및 테스트 베드 구축 방법

The algorithm design and the test bed construction method of processing for periodic delayed data

고상훈^{1*} · 송호진² · 금남호² · 유필중¹ · 오세권¹ · 김영성¹

¹LIG넥스원 PGM연구소

²국방과학연구소

Sang-hoon Koh^{1*} · Ho-jin Song² · Nam-ho Keum² · Pil-joong Yoo¹ · Se-kwon Oh¹ · Young-sung Kim¹

¹PGM Research and Development Lab, LIG Nex1 Co., Ltd., Korea

²Agency for Defense Development, Korea

[요 약]

유도탄 체계개발 기간 중 유도탄 기능 점검을 위하여 유도탄 종합점검장비를 제작하여 사용하는데, 장비 제작을 위하여 전원 및 통신 등에 대한 요구사항을 관리한다. 유도탄 종합점검장비 개발자는 유도탄과 통신하는 데이터의 신뢰성 확보를 위하여 제시된 통신 규격에 따라 소프트웨어를 구현한다. 구현 후 테스트베드를 구축하여 자체 성능평가를 수행하고 기존 장비를 이용하여 검증하는 프로세스로 진행하는데, 테스트 베드 구축 시 데이터의 주기적 지연 전송 특성을 반영하여야 한다. 본 논문은 주기적 지연 데이터 처리를 위한 테스트 베드 구축 방법에 대하여 기술하였고, 기존 설계된 알고리즘을 변경하여 성능을 비교 평가 하였다.

[Abstract]

The MATS(Missile Assembly Test Set) is manufactured and used to check the function of the missile during the period of development for the guided missile system, and the requirements for power and communication are managed for equipment production. The MATS developer implements software according to the proposed communication standard to guarantee the reliability of the data that communicates with the guided missile. The test bed is built and self-performance evaluation is performed after implementation. And the verification process is performed using the standard equipment. The characteristics of periodic delay for data transmission must be reflected when building a test bed. This paper describes a test bed construction method for data processing with periodic delay. Also This paper compares and evaluates the performance by changing the previously designed algorithm.

Key word : MATS(Missile Assembly Test Set), SDLC(Synctonized Data Link Control), Test-bed.

<http://dx.doi.org/10.12673/jant.2023.27.1.102>



This is an Open Access article distributed under the terms of the Creative Commons Attribution Non-Commercial License (<http://creativecommons.org/licenses/by-nc/3.0/>) which permits unrestricted non-commercial use, distribution, and reproduction in any medium, provided the original work is properly cited.

Received 13 January 2023; **Revised** 31 January 2023

Accepted (Publication) 20 February 2023 (28 February 2023)

***Corresponding Author** : Sang-hoon Koh

Tel: +82-31-5179-4189

E-mail: sanghoon.koh@lignex1.com

1. 서론

유도탄은 발사 이후 구성품의 전 기능이 동작하여 미션을 수행하는 원-샷 시스템으로, 운용 기간 중 발사 절차를 밟기 전까지 지상에서 충분한 점검을 통하여 유도탄의 신뢰도를 확보하는 것이 중요하다[1]-[2].

유도탄 점검을 위하여 유도탄 발사장비 모의 및 표적 모사를 할 수 있는 장비가 필요하고, 기능에 따라 모듈별로 나누어 설계 및 제작한다. 제작 이후 각 모듈 별 단위 테스트를 수행하고 시스템 통합을 통하여 장비를 구성한다. 이러한 장비 구성을 통틀어 유도탄 종합점검장비(이하 ‘점검장비’)라 한다[3]-[5].

점검장비는 무기체계 개발 시 해당 체계 검증을 위하여 사용되기에, 요구 항목에 따른 사양서를 작성하여 개발 기간 동안 점검장비의 형상 관리를 수행한다. 개발자는 해당 문서를 분석 후 각 기능을 구현할 수 있는 하드웨어 제작 및 소프트웨어 구현을 한다. 이후 점검장비 검증 프로세스에 따라 모듈 별 테스트 베드를 구축하여 점검장비 자체 성능 평가를 수행하고 기준장비를 통하여 성능을 입증한다.

그림 1은 점검장비의 구성과 유도탄 및 운용자 컴퓨터 간 인터페이스를 표현하였다. 먼저 운용자 컴퓨터로 점검장비의 시뮬레이션 모듈을 제어하여 점검에 필요한 모의 환경을 구성한다. 환경 모사가 완료되면 운용자 컴퓨터로 전원 공급 모듈을 제어하여 점검장비의 인터페이스 모듈로 유도탄에 전원을 공급한다. 유도탄의 부트-업 시간 이후 운용자 컴퓨터에서 점검 명령 데이터를 점검장비의 통신 모듈에 전송하면, 점검장비는 해당 데이터를 인터페이스 모듈을 통하여 유도탄에 전달한다. 유도탄에서 점검 명령에 대한 응답 데이터를 점검장비로 전송하면, 점검장비는 운용자 컴퓨터로 전달한다. 이후 운용자 컴퓨터의 요청에 따라 유도탄은 동작 상태 데이터를 주기적으로 전송하게 되고, 운용자는 해당 데이터를 분석하여 유도탄을 검증한다. 유도탄 통신과 전원 공급을 위한 전기적 인터페이스는 점검장비의 인터페이스 모듈과 유도탄의 배꼽(UMBilical) 구조물을 통하여 구성한다.

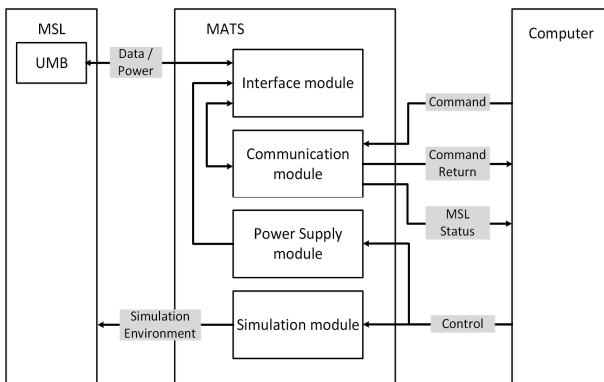


그림 1. 종합점검장비 연결도
Fig. 1. Connection of MATS and other equipment

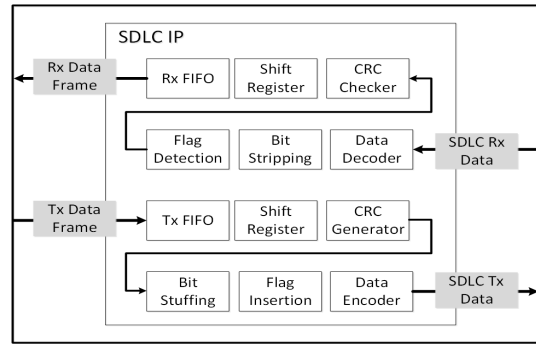


그림 2. SDLC 프로토콜을 적용한 데이터 흐름도
Fig. 2. Flow chart of data applying SDLC protocol

유도탄 점검 시 수신되는 동작 상태 데이터는 내부 구성품의 정보를 모두 포함하고 있어 데이터 프레임 사이즈가 크다. 해당 데이터 프레임이 점검장비에 주기적으로 전송되면 통신 모듈은 프레임 사이의 시간 간격 동안 데이터 처리를 완료해야하기 때문에 시간에 대한 제약사항이 발생한다[6]. 또한 해당 간격 동안 비주기 적으로 발생하는 점검 명령 데이터와 응답 데이터에 대한 처리가 요구되어 통신 모듈의 시간 제약은 더욱 커지게 된다. 이에 점검장비 개발 중 통신 모듈의 목표 성능 설정 시에는 요구되는 데이터 처리 양과 주기를 고려하여야 한다.

통신모듈이 수신한 신호는 SDLC(Synchronous Data Link Control) 프로토콜에 따라 데이터 프레임으로 변경된다. 이후 인터페이스 되는 장치에서 수신할 수 있게 하드웨어를 제공한다. 그림 2는 수신된 신호가 SDLC 데이터 프레임으로 바뀐 후 데이터 프레임으로 변경되는 과정을 나타낸다.

본 논문은 점검장비를 구성하고 있는 모듈 중 제약사항이 많은 통신 모듈을 대상으로 작성하였다. 2장에서는 통신모듈의 구성과 통신 기능 구현 및 알고리즘에 따른 기능 분석 결과를 기술하였다. 3장에서는 제작된 통신모듈의 검증을 위한 테스트 베드 구축 방법과 성능 측정 결과에 대하여 작성하였다.

II. 통신모듈 구성 및 기능분석

2-1 통신모듈 구성

유도탄과 운용자 컴퓨터 간 통신을 수행하기 위하여 그림 3과 같이 통신모듈을 구성하였다. 유도탄의 각 통신 채널에 대응하기 위하여 통신모듈의 트랜시버는 독립된 2개의 모듈을 사용하였고 운용자 컴퓨터와 통신하기 위하여 1개의 Ethernet 모듈을 사용하였다.(이하 각 모듈의 상세 품목은 보안상 생략하였다.)

SDLC(Synchronous Data Link Control) 프로토콜 컨트롤러 IP 코어(이하 ‘SDLC IP 코어’)는 상용 모듈을 사용하여 SDLC 데이터의 인코딩과 디코딩을 수행하였다.

FPGA(Field Programmable Gate Array)를 사용하여 해당 SDLC 기능을 구현하였고, 데이터의 저장 및 전송의 스케줄링을 위하여 MCU(Micro Controller Unit)를 사용하였다. MCU는 유도탄과의 SDLC 채널 송수신 데이터와 운용자 컴퓨터와의 Ethernet 송수신 데이터를 처리한다.

LAN 통신을 위하여 상용 모듈을 사용하였는데, 용도별로 소켓을 적용하여 운용자 컴퓨터와 통신을 한다. FPGA와 MCU의 어드레스와 데이터 라인을 통하여 XINTF(eXternal INTerFace)로 데이터를 전송하고 Ethernet 모듈의 인터럽트 라인은 MCU의 인터럽트 라인(INT#2)에 연결하여 운용자 컴퓨터에서의 데이터 수신을 확인한다.

XINTF를 위하여 MCU의 Zone0 메모리 영역은 SDLC IP 코어와, Zone6는 Ethernet 모듈과 연결한다. FPGA는 MCU의 메모리 영역과 연결된 각 Chip Select(CS#0, CS#6)를 참조하여 데이터 라우팅을 수행한다. SDLC IP 코어는 하나의 데이터 프레임 수신 완료 후 인터럽트 소스로 정의된 rdn(receive done interrupt) 신호선을 반전시키는데, 해당 신호선을 MCU의 인터럽트 라인(INT#0, INT#1, INT#2)에 연결하였다.

유도탄은 최대 10Mbps의 전송 속도로 데이터 프레임을 송신하고, 통신모듈의 SDLC IP 코어에서 데이터 프레임 저장 완료 후 생성되는 rdn 신호를 통하여 MCU의 Interrupt Service Routine으로 들어간다. 데이터 저장이 완료되면 MCU는 SDLC IP 코어에서 제공하는 수신 상태 레지스터를 변경하여 rdn 신호를 반전시켜 다음 데이터 프레임을 수신할 수 있는 상태로 변경한다. 다음 데이터 프레임이 들어오기 전에 MCU에서 수신 상태 레지스터를 변경하지 못하면 SDLC IP 코어에서 데이터를 수신할 수 없다.

운용자 컴퓨터에서 점검 명령 데이터가 들어오면 MCU는 Ethernet 모듈에서 데이터 수신 후 SDLC IP 코어에 데이터를 전송하여 해당 명령을 유도탄으로 전달한다.

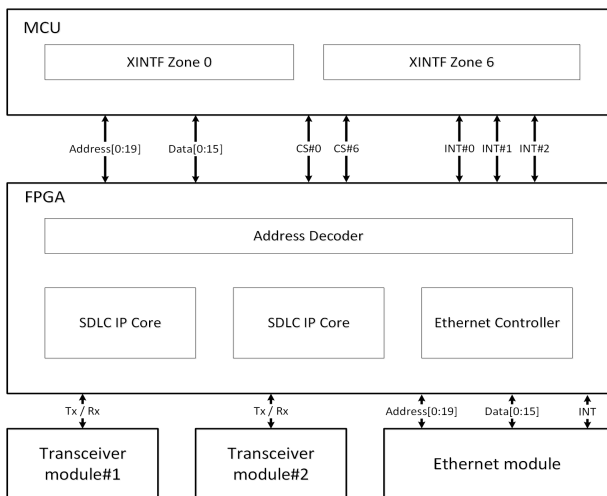


그림 3. 통신 모듈 형상
Fig. 3. Configuration of communication module

2-2 데이터 통신 시퀀스 차트 분석

운용자는 유도탄 점검을 위하여 점검 명령 채널을 통하여 유도탄 동작 상태 데이터 전송 요청을 수행하고, 점검 명령 데이터를 송신하여 동작 상태 데이터와 응답 데이터를 분석한다. 동작 상태 데이터는 처음 요청 후 전송 중지 요청이 있기 전까지 주기적으로 생성되어 점검장비로 전송된다. 그림 4는 유도탄 점검 상황에서 운용자 컴퓨터부터 유도탄까지의 데이터 흐름을 시퀀스 차트로 작성한 것이다.

운용자가 동작 상태 데이터 전송 요청 이후의 상황으로, 주기적으로 생성되는 동작 상태 데이터(StaTuS Data)가 수신 중 점검 명령(CoMmanD)과 응답 데이터(CoMmanD ReTurN)를 처리하는 과정이다. 운용자 컴퓨터가 점검 명령(COM port#1)을 보내면 점검장비는 해당 포트와 연결된 메모리(Wiznet_S1)에 저장한다. MCU는 Ethernet 모듈에 연결된 인터럽트를 통하여 수신 상태를 확인하고, 해당 메모리에 접근하여 RAM에 할당된 메모리 공간에 데이터를 저장한다. 저장 완료 후 External Interface를 통하여 유도탄 통신용 SDLC IP 코어(SDLC_MSL)에 데이터 프레임을 전송한다. SDLC IP 코어는 해당 프레임을 인코딩하여 트랜시버 모듈을 통하여 유도탄에 전송한다. 유도탄은 점검 명령 내용을 확인하여 응답 데이터를 생성하고 트랜시버 모듈을 통하여 점검장비로 전송한다. FPGA의 SDLC IP 코어에서 디코딩 후 인터럽트 신호를 발생시키면, MCU는 응답 데이터를 저장 후 FPGA의 SDLC IP 코어에 수신 완료하였음을 전달한다(ReCeiVe enable).

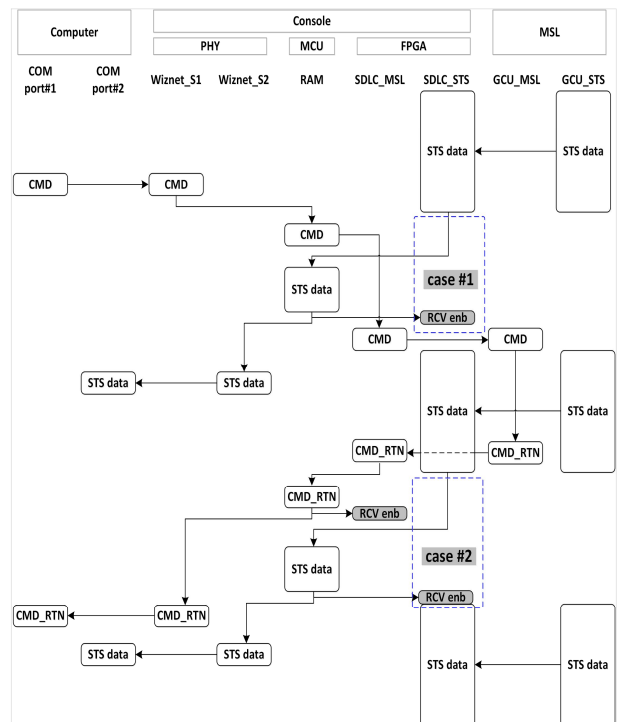


그림 4. 시퀀스 차트 분석
Fig. 4. Analysis for Sequence chart

이후 Ethernet 모듈의 해당 메모리(Wiznet_S1)에 저장하고 운용자 컴퓨터에 응답 데이터를 전송하여 점검 명령에 대한 응답을 송신한다. 동작 상태 데이터는 트랜시버 모듈을 통하여 해당 SDLC IP 코어(SDLC_STS)로 수신하고 MCU는 해당 데이터 프레임의 저장 후 Ethernet 모듈 메모리(Wiznet_S2)에 복사하여 운용자 컴퓨터의 해당 포트(COM port#2)로 전송한다.

점검장비는 SDLC IP 코어에서 동작 상태 데이터를 수신 중 운용자 컴퓨터에서 전송한 점검 명령 데이터나 응답 데이터를 수신한다. 이러한 경우 MCU는 순차적으로 데이터를 처리하고, 동작 상태 데이터나 응답 데이터 수신에 완료된 경우 SDLC IP 코어의 수신 상태 레지스터를 통하여 수신 완료되었음을 전달한다. 다만 동작 상태 데이터의 경우 다음 데이터가 들어오기 전에 MCU에서 수신 상태 레지스터를 변경하여야 SDLC IP 코어에서 다음 동작 상태 데이터를 수신할 수 있다.

운용자 컴퓨터에서 점검 명령 데이터 수신에 상태 확인 데이터 수신 종료 직전에 완료되면, MCU는 점검 명령 데이터를 처리하는 시간만큼 상태 확인 데이터에 대한 수신 완료 신호를 늦게 전달한다(case #1). 또한 응답 데이터 수신에 상태 확인 데이터 수신보다 먼저 완료되어도 같은 지연이 발생하고(case #2), 해당 두 개 케이스는 발생빈도가 낮지만 MCU에서 가장 가혹한 상황이다.

점검장비의 MCU는 1개의 core로 동작하기에 병목 현상이 발생하고 동작 상태 데이터 전송 주기 사이에 처리하여야 데이터 손실이 발생하지 않는다. 하지만 유도탄과의 통신 환경에 따라 데이터의 주기가 지켜지지 않는 경우 2번째 케이스(case #2)와 같이 다음 동작 상태 데이터 수신에 여유가 없는 경우가 발생한다. 이러한 환경을 극복하기 위하여 데이터 프레임 저장 조건을 변경하여 통신 환경 변화에 대응하여야 한다.

2-3 통신 처리 구조 변경

MCU의 데이터 저장 완료 시간을 앞당기기 위하여 표 1의 SDLC IP 코어가 제공하는 수신 상태 레지스터를 이용하였다.

표 1. SDLC IP core의 수신 상태 레지스터
Table 1. Receive status register of SDLC IP core

Bit	Description
rstat.7	Overrun error
rstat.6	Abort detect
rstat.5	Alignment error
rstat.4	CRC error
rstat.3	Receive done
rstat.2	Receive fifo not empty
rstat.1	Receive enable
rstat.0	Not used

rstat. 1부터 rstat. 7까지 데이터 프레임 수신 상태를 나타낸다. rstat. 7은 SDLC IP 코어가 최대 수신 가능한 크기를 초과하여 데이터가 수신되었을 때 설정되는 비트이고 rstat. 6은 송신측으로부터 오는 데이터 프레임이 멈춰 SDLC IP 코어의 데이터 수신에 중지되었음을 나타내는 비트이다. rstat. 5는 데이터 프레임의 워드가 16bit로 끝나지 않은 경우 설정되는 비트이고 rstat. 4는 0x7E와 CRC 워드를 제외한 데이터 프레임의 CRC 계산 결과가 송신측에서 전송한 CRC와 다르다면 설정되는 비트이다(CRC 계산 방법은 사전에 정의되어 있는 초기값과 계수를 이용하여 송수신단에서 동일하게 수행한다.). rstat. 3은 데이터 프레임 수신에 정상 완료되면 설정되는 비트이고 rstat. 2는 데이터 프레임이 SDLC IP 코어에 저장되어 rfifo 레지스터를 통하여 데이터 전송 가능함을 나타내는 비트이다. rstat. 1은 SDLC IP가 수신 가능 상태로 만들어 주는 비트로 rstat. 4 ~ rstat. 7의 에러를 모두 초기화시킨다.

해당 수신 상태 레지스터를 이용하여 데이터 프레임 복사 조건을 기존 rdn을 통한 인터럽트에서 rstat. 2인 “Receive fifo not empty” 비트를 확인하는 조건으로 변경하였다. 기존에는 SDLC IP 코어의 상태 확인 데이터 수신 완료 후 발생하는 rdn 신호를 기준으로 데이터 프레임을 복사하였는데, rstat. 2가 전환되는 조건으로 변경하여 데이터 프레임 저장 시점을 앞당겼다. 데이터를 버퍼에 저장할 때 FIFO에 데이터가 있는지 수신 상태 레지스터를 확인 후 FIFO에서 1바이트를 수신하고, 데이터 프레임이 끝날 때 까지 수신 상태 레지스터 확인과 FIFO 데이터 수신하는 것을 반복한다.

데이터 프레임을 저장하는 동안 수신 상태 레지스터 1바이트를 읽고 FIFO에서 1바이트를 저장하기에, 전의 구조에 대비하여 2배 이상 데이터를 읽어 MCU의 자원 손실이 있지만 유도탄에서 전송하는 속도보다 XINTF의 속도가 빨라 구조 변경이 가능하다.

2-4 변경 알고리즘 분석

변경된 조건으로 그림 5와 같이 시퀀스 차트를 재작성 하였다. MCU는 IDLE 상태에서 수신 상태 레지스터를 계속 수신하여 조건과 일치하는지 확인한다. 유도탄에서 데이터 프레임이 수신되어 SDLC IP 코어의 rstat. 2 비트가 전환되면 SDLC 데이터 1바이트를 수신하고, 데이터 프레임 수신에 완료될 때 까지 수신 상태 레지스터와 데이터를 번갈아 수신한다. 수신이 완료되면 수신 상태 레지스터를 통하여 수신 완료되었음을 SDLC IP 코어에 전달하여 새로운 데이터를 수신할 수 있는 상태로 변경한다. 동작 상태 데이터 수신 중 점검 명령 데이터가 도착하면 MCU는 먼저 수신 중이던 데이터 프레임을 처리 후 수신 상태 레지스터를 변경을 마치고, 메모리에 대기 중이던 점검 명령 데이터를 수신한다. 기존과 달리 동작 상태 데이터를 저장하는 도중 점검 명령 데이터 혹은 응답 데이터가 동작 상태 데이터와 동시에 들어와도 MCU는 상태 확인 레지스터가 먼저 바뀐 데이터 프레임을 먼저 처리 후 다른 두 개 데이터를 처리한다.

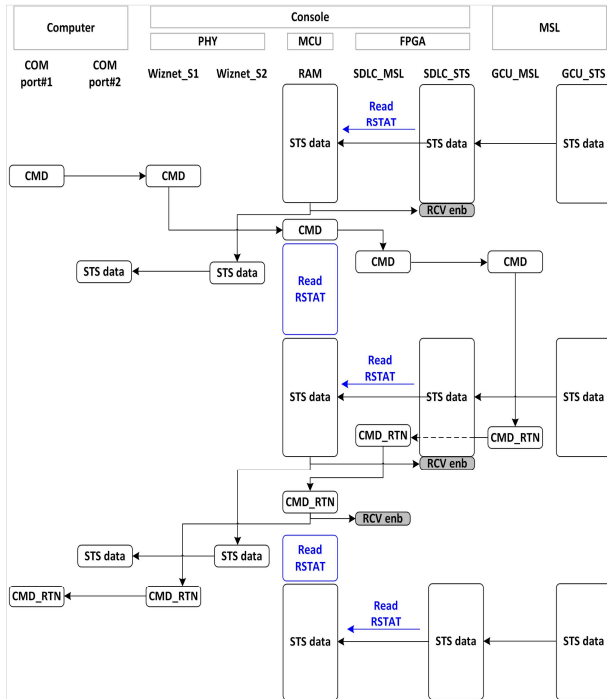


그림 5. 수정 후 시퀀스 차트
Fig. 5. Sequence chart after modify

III. 테스트 베드 구축 및 성능 측정

3-1 테스트 베드 구축

그림 6은 데이터 수신에 따른 MCU의 상태를 나타낸다. 통신 모듈의 처리 속도에 대한 목표치는 동작 상태 데이터의 수신 간격(Message Gap Time)과 점검 시나리오에 따른 점검 명령 데이터 수신 및 응답 데이터 수신 상황을 고려하여 설정하였다 [6]. 유도탄이 동작 상태 데이터를 전송하지 않는 시간(Inter Message Gap Time, 이하 t_{IMGT}) 동안 MCU가 SDLC IP 코어에서 데이터 수신(t_1) 완료 후 다음 데이터가 들어오기까지 남은 시간(t_2) 안에 수신 상태 레지스터를 변경하여 주어야 rcv 신호(rcv#1)가 초기화되면서 다음 동작 상태 데이터를 받을 수 있다(응답 데이터는 rcv#2 초기화 수행). SDLC 프로토콜로 데이터 전송 시 opening flag 및 closing flag 인 0x7E(0b0111110)과 데이터를 구분하기 위하여 5개의 비트가 연속으로 1이 쓰이면 다음 데이터에 의무적으로 0을 넣어 0x7E와 구분을 한다. 이를 bit stuffing 이라 하고, 데이터가 1bit 추가되어 데이터 길이와 이에 따른 전송 시간이 늘어난다. 전송에 가장 가혹한 상황을 만들기 위하여 모든 데이터를 0xffff로 가정하면 bit stuffing에 의하여 80bit 당 16bit가 추가되어 전송 데이터가 총 96bit가 된다. 이에 요구되는 데이터 프레임의 길이에 125%를 곱하여 데이터 전송 속도를 나누어 t_{MGT} 에서 빼주면 MCU 처리 속도 목표 성능치인 t_{IMGT} 를 구할 수 있다.

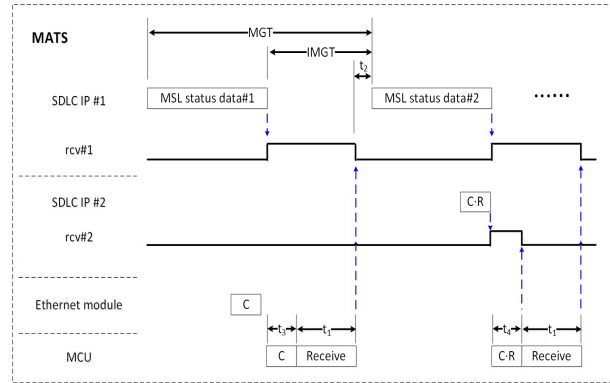


그림 6. 통신 모듈 타이밍 다이어그램
Fig. 6. Timing diagram of communication module

동작 상태 데이터 수신 중 운용자 컴퓨터에서의 점검 명령 데이터(Command)나 유도탄에서의 응답 데이터(Command Return)가 수신되는 경우가 발생하지만 데이터가 비주기적이고 다음 데이터와의 간격이 동작 상태 데이터보다 훨씬 크기에 t_{IMGT} 에서 t_1 을 제외한 시간에 처리하여도 데이터 수신 가능하다. 다만 해당 데이터들이 동작 상태 데이터보다 먼저 점검 장비에 도착하여 수신하면, 점검 명령 수신 시간(t_3) 혹은 응답 데이터 수신 시간(t_4)만큼 t_2 가 줄어들게 된다. 테스트 베드를 구축하여 t_1 과 t_3 및 t_4 를 측정하여 $t_1 + t_3$ 혹은 $t_1 + t_4$ 의 최대값이 t_{IMGT} 보다 짧아야 한다. 식 1은 동작 상태 데이터의 전송 속도 및 데이터 길이 조건 중 가장 가혹한 상황으로 설정하여 t_{IMGT} 를 계산하였다.

$$t_{IMGT} = t_{MGT} - \frac{(L \times 125\%)}{S} \quad (1)$$

L : Length (데이터 프레임 길이, bit)

S : Speed (데이터 전송 속도, bit / second)

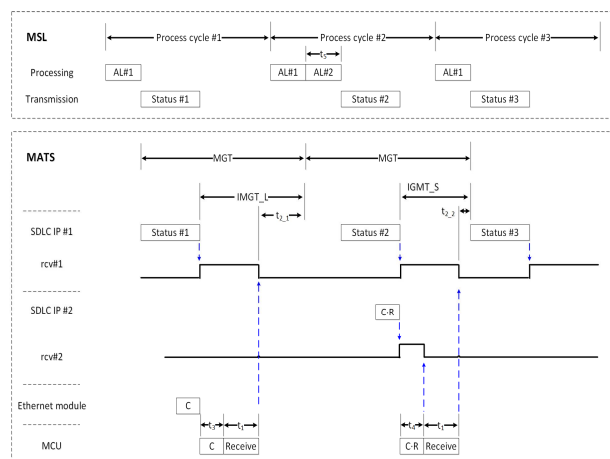


그림 7. 지연 데이터 송수신 타이밍 다이어그램
Fig. 7. Timing diagram of communication for delayed data

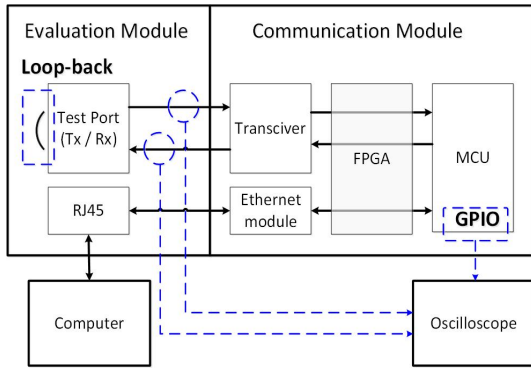


그림 8. 통신 모듈의 테스트 베드
Fig. 8. Test bed for communication module

추가로 테스트 베드를 구축하기 위하여 고려할 사항은 동작 상태 데이터의 주기적 지연 특성이다. 그림 7은 동작 상태 데이터(Status) 전송의 지연 현상을 나타낸다. 유도탄의 경우 수행 알고리즘(ALgorithm#2)의 주기성으로 연산 주기 내에서 동작 상태 데이터 전송 시간이 바뀌게 된다. 이로 인하여 점검장비의 t_{IMGT} 가 줄어든다(그림의 $IMGT_Short$). 해당 사유로 목표치인 t_{IMGT} 를 조정할 필요가 있고, 주기적으로 연산하는 알고리즘(ALgorithm#2)의 처리시간(t_5)를 계측하여 반영하여야 한다. 또한 t_1 과 t_3 및 t_4 를 측정하여 $t_1 + t_3$ 혹은 $t_1 + t_4$ 의 최대값이 t_5 가 반영된 $t_{IMGT}(t_{target})$ 보다 짧아야 한다. 해당 사항을 반영하여 목표 성능을 t_{IMGT} 에서 t_5 만큼 감소한 t_{target} 으로 설정하여야 한다.

그림 8은 통신모듈의 수신 성능을 측정하기 위한 테스트 베드이다. 개발모듈을 제작하여 통신 신호를 테스트 포트에 뽑아내고 치구를 이용하여 통신 라인을 루프-백 시켰다. 개발 모듈의 RJ45 연결기를 통하여 운용컴퓨터로 통신 개설을 한 후 모의 데이터를 전송하였고, 계측을 위하여 트랜시버 송수신단을 오실로스코프에 연결하였다. MCU에서 데이터 수신 후 GPIO 신호를 반전시켜 외부에 연결된 오실로스코프로 함수 처리 속도를 측정하였다[7].

3-2 성능 측정

통신모듈에서 동작 상태 데이터를 제외하고 유도탄과 운전자 컴퓨터에서 동작 명령 및 응답 데이터가 비동기식으로 수신된다. 이로 인하여 데이터 수신에 따른 인터럽트가 비동기식으로 발생할 수 있어 태스크 수행 중 발생할 수 있는 오류를 방지하기 위하여 더버퍼링이 필요하다[8]. 데이터 수신 분야에서 효율성이 입증된 링버퍼를 해당 통신 모듈의 MCU에 채택하여 안정적인 버퍼링을 할 수 있게 구현하였다[9].

이후 테스트 베드를 구성하여 기존 통신 알고리즘과 변경한 알고리즘을 업로드 하여 처리 시간을 측정하였다. 측정에는 Rohde&Schwarz사의 오실로스코프 RTO2024(최대 10 GS/s를 지원)를 이용하였고, 결과는 표 2, 3과 같다.(표의 백분율 값은

t_{target} 을 기준으로 각 데이터 처리 시간을 나타낸 것이다. 이는 실제 유도탄 데이터 통신 기준에 대한 수치를 표현하지 않기에 위함이다. 이후 나오는 그림에 대하여서도 축의 단위에 대하여 생략하도록 한다.)

표 2. 기존 데이터 수신에 따른 프로세스 타임
Table 2. Process time for exciting algorithm

Process	Time	Direction
Receiving status data(①)	657.1 %	MSL → Console
Receiving command(②)	30.4 %	COM → Console
Receiving command return(③)	49.3 %	MSL → Console

표 3. 새로운 조건에서의 프로세스 시간
Table 3. Process time after changing condition

Process	Time	Direction
Receiving status data(④)	2.9%	MSL -> Console
Receiving command(⑤)	30.4%	COM -> Console
Receiving command return(⑥)	0.2%	MSL -> Console

표 2의 경우 기존 알고리즘을 적용했을 때의 결과이다. 동작 상태 데이터 수신에 목표치인 t_{target} 을 만족하지 못하는 것을 확인하였다.

표 3은 변경한 알고리즘을 적용했을 때의 결과이다. 동작 상태 데이터 수신과 응답 데이터 수신에 각각 657.1%에서 2.9%로, 49.3%에서 0.2%로 줄어든 것을 확인하였다. 동작 상태 데이터 수신 시간(④)과 점검 명령 데이터 수신 시간(⑤)의 합($t_1 + t_3$)과 응답 데이터 수신 시간(⑥)과의 합($t_1 + t_4$) 중 최대값은 t_{target} 기준 33.3%로, 가장 가혹한 상황이 발생하여도 MCU에서 모든 데이터 처리가 가능함을 확인하였다.

3-3 점검장비 적용 및 확인

점검장비 평가를 위하여 유도탄과 동일한 성능을 갖춘 기준탄을 이용하였고, 운용컴퓨터에 점검별 시나리오를 구현하여 점검장비의 동작 이상 유무를 확인하였다. 그림 9와 그림 10은 기존 방식과 변경한 방식으로 계측한 결과이다.

그림 9의 2번 채널(Ch#2)은 RS422 트랜시버로 수신된 동작 상태 데이터이고 1번 채널(Ch#1)은 MCU가 데이터 프레임 수신 처리 후 MCU의 GPIO를 반전시킨 것이다. RS422 데이터 프레임 간격이 좁은 경우 두 번째 데이터 프레임 저장에 대한 반전 신호가 확인되지 않았다.

그림 10의 2번 채널(Ch#2)은 동작 상태 데이터 프레임이 트랜시버 모듈에 수신된 것이고 1번 채널(Ch#1)은 해당 데이터 프레임을 저장하는 MCU의 상태를 GPIO로 출력한 신호이다. 4

번 채널(Ch#4)은 운전자 컴퓨터에서 점검장비로 점검 명령 데이터 수신 후 점검장비에서 유도탄으로 전송되는 신호이고 3번 채널(Ch#3)은 유도탄에서 점검장비로 전송되는 응답 데이터 수신에 대한 신호이다. 동작 상태 데이터 수신 간격이 좁아도 MCU는 정상적으로 동작하는 것을 확인하였다.

데이터 손실이 없는지 확인하기 위하여 패킷 분석 툴인 Wireshark를 사용하였다[10]. 그림 11과 그림 12는 기존 방식과 변경된 방식으로 알고리즘을 적용하여 일정 기간 동안 수신한 동작 상태 데이터의 수신 간격 시간에 대한 분포도이다. 각 그림의 Section 1은 동작 상태 데이터의 주기에 수신된 데이터 영역이고, Section 2는 전송 주기 2배의 데이터 영역이다. 데이터 누락이 생기면 Section 2 영역 등 MGT의 배율로 데이터 수신 기록이 발생한다.

그림 11의 Section 2에 데이터가 있는 것은 이전 데이터와 현재 데이터 사이의 간격이 벌어졌다는 것으로 데이터 프레임의 누락을 의미한다.

그림 12에서 보면 Section 1에 모든 데이터가 분포되어 있고, 점검장비가 유도탄의 모든 동작 상태 데이터를 빠짐없이 수신함을 보여준다.

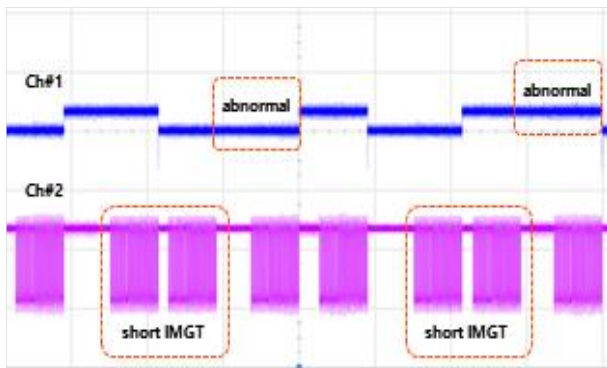


그림 9. 기존 알고리즘 통신 신호 계측
Fig. 9. Measurement for signal for existing algorithm

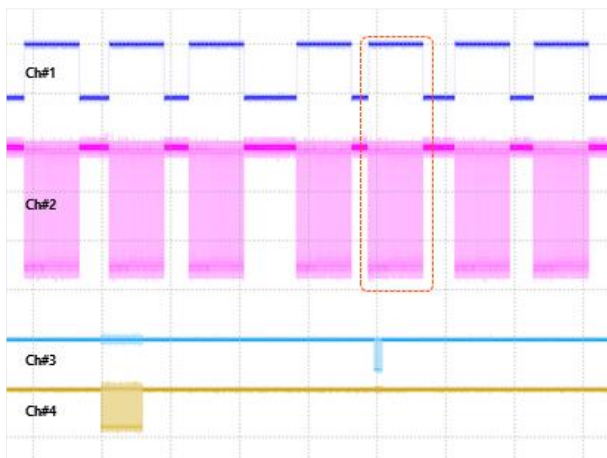


그림 10. 통신 신호 계측
Fig. 10. Measurement for missile communication

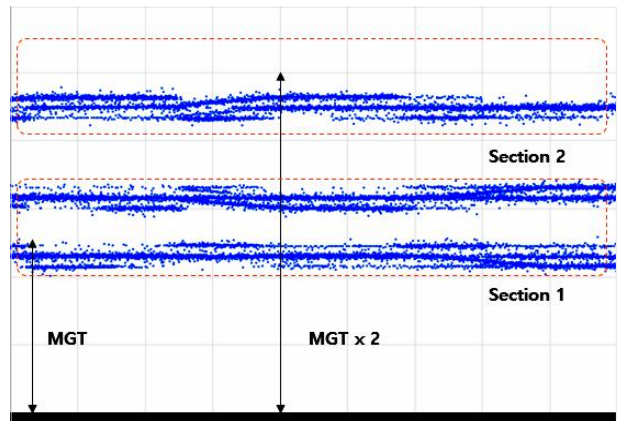


그림 11. 기존 알고리즘 동작 상태 데이터 수신 간격 분포도
Fig. 11. Time graph for existing algorithm

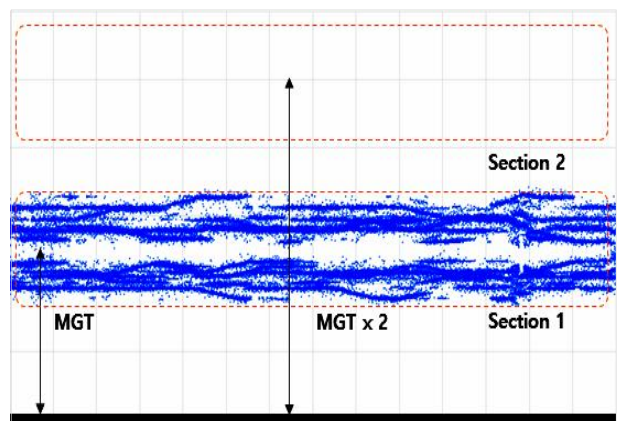


그림 12. 동작 상태 데이터 수신 간격 분포도
Fig. 12. Time graph for status data

IV. 결 론

본 논문은 점검장비의 통신 모듈에 대한 설계부터 평가까지의 과정을 기술하였다. 먼저 기존 알고리즘의 데이터 처리 방식과 시퀀스 차트 분석 결과를 살펴보고 통신 상황에 따라 제약 사항이 발생할 수 있음을 예측하였다. 예상되는 상황을 보완하고자 제공되는 정보를 기반으로 데이터 저장 시점을 앞당겼고 시퀀스 차트 분석으로 이전 제약사항이 발생하지 않음을 확인하였다. 이후 테스트 베드 구축 시 목표치 설정 방법에 대하여 기술하였고, 주기적 데이터 전송 지연 상황을 고려하여 목표치 설정을 변경하여 테스트 베드를 보완하였다.

테스트 베드를 통하여 기존 데이터 처리 방식과 성능을 비교 분석하였고, 주기적 데이터 전송 지연 환경에서 기존 방식이 성능 만족하지 못함을 확인하였다. 이후 기준탄을 이용하여 점검장비 성능 평가를 수행하였고 데이터 누락 없이 수신함을 확인하여 점검장비 검증을 완료하였다.

본 논문에서 제안한 방법은 통신 규격의 최대 전송 속도보다 최소 2배 이상 빠른 XINTF를 제공하는 모듈을 적용하여 주기

적 데이터 전송 지연으로 인한 병목 현상을 극복하고 모든 데이터를 누락 없이 처리할 수 있다. 이후 점검장비 개발에 더욱 빠른 처리 속도가 요구되는 경우 PCI-express(5.0세대 기준 32GT/s)가 적용된 모듈을 적용하여야 한다[11].

References

- [1] B. S. Kim, K. S. Lee, D. S. Kim, K. S. Moon, "Certified Missile Rounds Concepts Using Modeling and Simulation", *JKSS*, Vol. 18, No. 4, pp. 95-105, Dec. 2009.
- [2] Y. H. Lee, J. Y. Kim, K. S. Moon, K. S. Lee, "A Study on Stockpile Reliability Program for Effective Life Cycle Management of Domestically Developed Missile", *Journal of The Korea Association of Defense Industry Studies* Vol. 28, No. 1, 2021.
- [3] J. L. Anderson, "High Performance Missile Testing(Next Generation Test Systems)", *Proceedings AUTOTESTCON 2003. IEEE Systems Readiness Technology Conference.*, pp. 19-27, 2003.
- [4] Ron Drees, Neal Young, "Role of BIT in Support System Maintenance and Availability", *IEEE Aerospace and Electronic Systems Magazine*, Vol. 19, No. 8, Aug. 2004.
- [5] Mark Klafter, Diana Eshelman, "Honeywell's Missiles Universal Test Set: An Automated, WindowsTM-Based, Test Station Providing Simultaneous System Testing Capability", *International Automatic Testing Conference, AUTOTESTCON*, pp. 11-18, Sep. 2003.
- [6] K. P. Kim, "Analysis and Implementation of High Speed Data Processing Technology using Multi-Message Chain and Double Buffering Method with MIL-STD-1553B", *KIMST*, Vol. 16, No. 4, pp. 422-429, Aug. 2013.
- [7] Liu Zhaoqing, Wang Chu, Zhang Yigang, "Design and Implementation of a Certain type of missile timing sequence testing systems", *2013 IEEE 11th International Conference on Electronic Measurement*
- [8] C. H. Koo, "Method of data processing through polling and interrupt derive I/O on device data", *KSAS*, Vol. 33, No. 9, pp. 113-119, Sep. 2005.
- [9] M. G. Song, H. R. Kim, "Implementation of Ring Buffer based Massive VLBI Data Stream Input/Output over the Wide Area Network", *KIECS*, Vol. 14, No. 6, pp. 1109-1120, Dec. 2019.
- [10] <https://www.wireshark.org>
- [11] Kun Cheng, Weiyue Liu, Qi Shen and Shengkai Liao, "Design and Implementation of High-throughput PICE with DMA Architecture between FPGA and PowerPC", *JOURNAL OF IEEE TRANSACTIONS ON NUCLEAR SCIENCE*, Sep. 2018



고 상 훈 (Sang-Hoon Koh)

2011년 2월 : 고려대학교 전기전자컴퓨터공학과 (공학석사)
2011년 1월 ~ 현재 : LIG넥스원 유도무기연구소 수석연구원
※관심분야 : 유도무기, 정보통신, 임베디드시스템, 체계공학



송 호 진 (Ho-jin Song)

2017년 2월 : 한국과학기술원 전기 및 전자공학부 (공학박사)
2017년 3월 ~ 현재 : 국방과학연구소 선임연구원
※관심분야 : 유도무기, 정보통신



금 남 호 (Nam-Ho Keum)

2017년 2월 : 고려대학교 컴퓨터 정보통신공학과 (공학석사)
2017년 9월 ~ 현재 : 국방과학연구소 선임연구원
※관심분야 : 유도무기, 정보통신



유 필 중 (Pil-joong Yoo)

2009년 8월 : 경희대학교 동서의료공학과 (공학사)
2011년 8월 : 경희대학교 동서의료공학과 (공학석사)
2021년 11월 ~ : LIG 넥스원 유도무기연구소 선임연구원
※관심분야 : 유도무기, 임베디드 시스템



오 세 권 (Se-kwon Oh)

2011년 2월 : 명지대학교 정보통신공학과 (공학사)
2013년 2월 : 명지대학교 정보통신공학과 (공학석사)
2013년 1월 ~ 현재 : LIG넥스원 선임연구원
※관심분야 : 유도무기, 탄 체계, 체계공학, 소프트웨어공학



김 영 성 (Young-sung Kim)

2015년 2월 : 한양대학교 컴퓨터공학과(공학사)
2017년 2월 : 한양대학교 컴퓨터공학과 (공학석사)
2017년 1월 ~ 현재 : LIG넥스원 선임연구원
※관심분야 : 유도무기, 탄체계, 제어알고리즘