

성하고 시뮬레이터 PC와 게이트웨이에서의 제어 명령을 받아 수행한다. LabVIEW RTOS는 데이터의 입력, 데이터 출력의 모든 처리를 실시간으로 수행하며 데이터의 입출력을 위해 GPIB, LAN, RS-232, RS-422, 아날로그/이산 입출력 인터페이스를 사용한다. 그리고 Time code generator에서 보내는 시간 정보를 서버로부터 받아서 실시간 데이터 처리를 위한 시간 동기화를 이룬다.

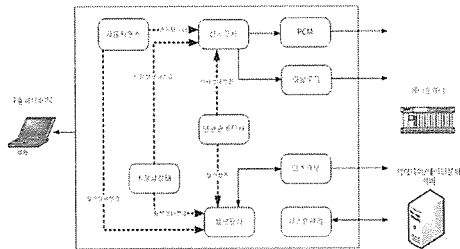


그림 3 시뮬레이터 기능도

2.2.2 게이트웨이 PC와 게이트웨이

발사관제시스템 프로토타입에서 게이트웨이는 시뮬레이터에서 전송되는 데이터를 일정 포맷형태로 재구성하여 서버로 전송하고 게이트웨이 PC가 Console의 기능으로 서버에 보낸 명령을 서버로부터 받아서 시뮬레이터에 전송하는 기능을 가진다. 그림 4는 게이트웨이 기능도를 나타낸다. 게이트웨이 PC에서 LabVIEW를 사용하여 작성된 게이트웨이 프로그램은 LabVIEW RTOS 운영프로그램에 TCP/IP를 통해 다운로드되어 게이트웨이로서의 역할을 다음과 같이 수행한다.

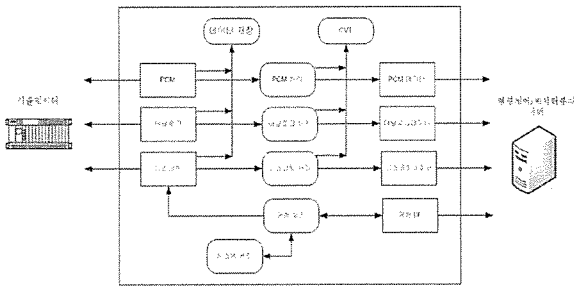


그림 4 게이트웨이 기능도

시뮬레이터 PC로부터 주어지는 PCM 데이터 포맷에 맞추어 시뮬레이터로부터 받은 PCM 데이터를 디코딩하며, 이 데이터에 시간정보, 엔지니어링 단위 등을 추가하는 작업을 한다. 게이트웨이는 서버를 통해 입력된 게이트웨이 PC(콘솔의 임무를 수행 시)의 명령을 수행하고 시뮬레이터와 유사하게 LabVIEW RTOS는 데이터의 입력, 게이트웨이 기능, 데이터 출력의 모든 처리를 실시간으로 수행하며 데이터의 입출력을 위해 GPIB, LAN, RS-232, RS-422, 아날로그/이산 데이터 입출력 인터페이스를 사용한다. 또한 서버에서 보내는 GPS 시간으로 시간 동기화를 이룬다.

2.2.3 데이터 분배 서버

그림 5는 데이터 분배 서버 기능도를 나타낸다. 데이터 분배 서버는 발사체로부터 측정 되는 End Item의 압력, 레벨, 온도, 가속도, 변형률 등의 정보 및 시스템 상태를 수집하고 분석하여 분배하는 기능 및 게이트웨이와 자신의 현 상태를 지속적으로 감시하다 이상 발생을 감지하면 이상 발생 보고서를 작성하고, 이를 운영자와 명령 제어 서버의 시스템 감시 기능에 보고하는 기능을 수행한다.

그리고, 지속적으로 수집되는 데이터에 대한 저장 기능을 수행한다. 데이터 분배 서버는 시뮬레이터로부터 발생하는 PCM 데이터와 게이트웨이에서 처리 및 수정된 데이터 그리고 데이터 분배 서버에서 처리된 측정 데이터를 발생순서와 종류에 따라 모두 저장하는 역할을 수행한다. 이는 발사 시험 종료 후 분석을 위한 기본 자료로 제공된다. 이처럼 데이터 분배 서버는 발사관제 시스템 프로토타입을 구성하는 모든 데이터를 분석/분배하는 역할을 수행하게 된다.

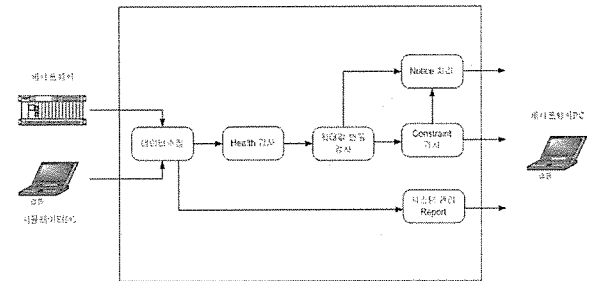


그림 5 데이터 분배 서버 기능도

2.2.4 명령 제어 서버

그림 6은 명령 제어 서버 기능도를 나타낸다. 명령 제어 서버는 가상 발사체의 시험 및 발사 과정을 제어하기 위한 수동/자동 제어와 발사관제시스템 프로토타입의 시간 동기화를 위한 소스 역할을 담당한다. 그림 6은 명령 제어 서버 기능도를 나타낸다. 가상 발사체의 시험 및 발사 시퀀스 제어는 외부 전력 공급 명령이나 산화제 및 연료 주입 절차 등의 발사 과정에서 필요로 하는 절차를 수행한다. 이들의 명령은 운영자에게 의한 수동 제어와 사전에 정의된 명령 시퀀스 시나리오에 따른 자동 제어로 나뉘질 수 있다. 수동/자동 제어 외에 명령 제어 서버는 명령에 대한 인증 및 허용 상태 유무를 확인 하는 명령 검증 기능, 그리고 발사 시험 및 발사 과정 시나리오에서 현재 수행하고자 하는 명령의 적합성을 확인 하는 기능 등이 명령 제어 서버를 통해 수행된다. 이러한 기능 처리 후 게이트웨이로 명령을 전달한다.

시스템 시간 동기화는 발사관제시스템 프로토타입을 구성하고 있는 게이트웨이, 명령 제어 서버, 데이터 분배 서버, 시뮬레이터/시뮬레이터 PC, 그리고 게이트웨이 PC의 시간을 동기화하기 위한 시간 동기화 신호를 발생시키고 각 서브시스템으로 동기화 정보를 IRIG-B로

제공하는 역할을 수행한다.

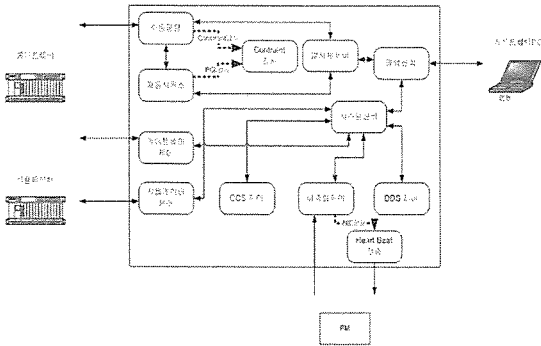


그림 6 명령 제어 서버 기능도

3. 발사관제시스템 프로토타입 설계

발사관제시스템 프로토타입은 UML(Unified Modeling Language)을 이용하여 객체지향 분석 설계 방법으로 설계하였다. UML은 객체지향 분석(Analysis)과 설계(Design)를 위한 모델링 언어로 시스템의 기능, 요구사항, 상호 연관관계 등을 가장 명확히 표현 할 수 있다. 설계 과정에서 유스케이스, 시퀀스, 클래스 다이어그램 등을 산출하였으며 UML툴은 Borland 사의 Together를 이용하였다. LabVIEW의 경우 LabVIEW GOOP(Graphical Object-Oriented Programming) 모듈을 이용하여 객체지향 프로그래밍을 수행하였다.

4. 발사관제시스템 프로토타입 실시간 데이터 처리

4.1 하드웨어 운용 및 단위 테스트

발사관제시스템 프로토타입에서의 시뮬레이터와 게이트웨이는 NI(National Instruments)사의 PXI 모듈들이 장착되어 있다. 표 1은 시뮬레이터와 게이트웨이를 구성하는 PXI 모듈들에 대해 나타나 있다. 서버는 Sun Blade 2000으로 UltraSPARC-III 1.2 GHz 프로세서, 1GB 메모리, 및 74GB의 하드디스크로 구성된다.

LabVIEW에서 데이터 처리과정 중 버퍼의 양을 체크하는 Single Scan 기능을 이용하여 메모리에 저장되는 데이터 bit를 체크하여 실시간적으로 데이터를 처리하는지를 확인하였다. 1000 samples/sec로 10000번 데이터를 수집하였을 때 버퍼에는 0~1bit만이 남는 것을 알 수 있었으며 실시간 데이터 처리 시 버퍼를 거의 사용하지 않는다는 것을 확인하였다.

4.2 소프트웨어 운용 및 단위테스트

실시간(Real Time) 데이터 처리의 개념은 시스템에서 데이터 처리에 대한 시간을 정확하게 규정한 것으로 OS 환경에서 외부의 인터럽트 혹은 이벤트에 대한 구애를 받지 않고 정확한 데이터 처리 시간을 보증해야 한

다. 또한 최대의 성능과 정확성을 가져야 하며 루프타임을 벗어난 시간(Jitter)을 최소화 할 수 있어야 한다. 발사

표 1 시뮬레이터, 게이트웨이 구성 모듈

NI PXI-1045 샤시	아래의 모듈들과 카드들을 설치하게 된다.
NI 8186RT PXI embedded controller	RTOS(Real Time OS)가 설치되어 실시간 신호처리를 담당한다.
NI PXI-6070E Multifunction I/O 모듈	시뮬레이터에서 전송하는 아날로그 신호를 게이트웨이에서 수신하는 모듈이다.
NI PXI-6704 Analog Output 모듈	시뮬레이터에서 아날로그 신호를 출력하는 모듈이다.
NI PXI-6527 Digital I/O 모듈(2)	시뮬레이터/게이트웨이 사이 이산(離散) 신호 송, 수신 모듈이다.
NI PXI-6608 Timing I/O 모듈	시스템간의 시간 동기화를 위하여 Time서버로부터 IRIG-B신호를 입력받는 모듈이다.
NI PXI-8423 RS-422 interface 모듈	시뮬레이터에서 생성하는 PCM 데이터를 게이트웨이에 전송하기 위한 인터페이스 모듈이다.
GPIB 인터페이스	게이트웨이에서 DC Power Supply를 제어하기위한 인터페이스이다.
Ethernet 카드	네트워크를 통하여 LabVIEW VI를 다운로드 받고 시뮬레이터/게이트웨이에서 이루어지는 데이터 처리를 GUI에서 볼 수 있도록 데이터를 보내 준다.

관제시스템 프로토타입을 구성하는 PXI 모듈들과 호환되는 LabVIEW Real Time Operation System(RTOS) 환경 하에서 임베디드 LabVIEW VI(Virtual Instrument) 프로그램을 개발하여 시험평가 하였다.[6] 표 2는 시뮬레이터와 게이트웨이에 대한 소프트웨어적 실시간 처리 결과를 보여 준다. 루프를 1000번 수행하게 하여 1000 Hz를 발생하게 하였을 때 그 정밀도를 LabVIEW RTOS 환경과 Window OS 환경 하에서 비교 하였다. LabVIEW RTOS 환경과 Window OS 환경에서의 비교 값을 보게 되면 Jitter가 Window OS환경이 LabVIEW RTOS 환경보다 200배나 크다는 것을 알 수 있으며 루프율 및 표준편차도 LabVIEW RTOS 환경이 우수하다는 것을 알 수 있다.

표 2. 소프트웨어적 실시간 처리 결과

	OS	배열 크기	loop rate mean (Hz)	Max Jitter (us)	표준 편차 (us)
시뮬레이터	LabVIEW RT	1×1000	999.3	0.6	0.18
	Windows	1×1000	996.8	129.6	9.47
게이트웨이	LabVIEW RT	1×1000	999.3	0.6	0.19
	Windows	1×1000	996.8	120.6	11.54

5. 발사관제시스템 프로토타입의 적용 결과

5.1 데이터 처리 결과

표 3은 PCM 데이터 포맷 형식을 나타내며 표 4, 5는 각 시스템에서 PCM 데이터 및 아날로그/이산 데이터 처리 시간을 나타낸다. SSR(System Synchronization Rate)은 각 시스템의 데이터 처리 시간을 의미하며 기준 처리값을 10ms로 설정하여 데이터 처리를 수행 하였다.[7] PCM 데이터 처리 시간은 초당 200 Major Frame 전송 시 320 kbps의 bit rates를 가진다.[8]

표 3 PCM 데이터 형식

항 목	규 격
ID0	0~19
ID1	0~3
Minor Frame /Major Frame	4
Words/Minor Frame	50
Bits/Word	8

5.2 성능 향상을 위한 제안

4.1절, 4.2절의 시뮬레이터, 게이트웨이에 대한 단위 테스트 결과를 보면 실시간 데이터 처리

능력이 우수함을 알 수 있었다. 5.1절의 발사관제시스템 프로토타입의 데이터 처리 결과를 살펴보면 아날

표 4 PCM 데이터 처리 시간

System	Data Processing(ms)
시뮬레이터	20
게이트웨이	10
데이터분배서버	22.3
명령제어서버	10

표 5 아날로그/이산 데이터 처리 시간

System	Data Processing(ms)
시뮬레이터	10
게이트웨이	10
데이터분배서버	10
명령제어서버	10

로그/이산 데이터 처리 시간은 SSR 10ms를 만족하는 것을 알 수 있으나 PCM 데이터 처리는 SSR 10ms로 처리되지 못한 것을 알 수 있다. 발사관제시스템 프로토타입에서 PCM 데이터 처리의 경우 실제 발사체의 PCM 전송 속도는 발사체 1단에서 1.5Mbps, 2단에서는 0.7Mbps를 만족해야 한다.[9] 그러나 현재 개발된 발사관제시스템 프로토타입에 사용된 PCM 전송을 위한 PXI-8423은 최대 460kbps 전송 속도 지원하고 게이트웨이는 320kbps 속도

를 가진다. 따라서 발사체에서 전송되는 PCM 데이터를 모사하지는 못한다. KSLV-I PCM 데이터의 모사를 위해서는 PCM 전용 하드웨어를 통해 데이터 전송속도를 향상시킬 수 있다.[10] 표 6은 PCM 전용 하드웨어의

표 6. PCM 전용 하드웨어 성능

전송 속도	최대 20 Mbps
Word Size	3 ~ 16 bit
Minor Frame length	2 ~ 8191 word
Major Frame length	최대 256 Minor Frame
동기화 코드	최대 64 bit

전송속도를 보여 준다.

그리고 개발된 발사관제시스템 프로토타입은 발사체 모사 데이터를 하나의 게이트웨이에서 다중처리를 하게 되어 처리 및 데이터 전송에 시간이 많이 걸리는 것을 알 수 있다.

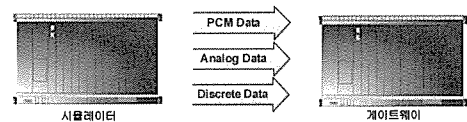


그림 7 발사관제시스템 프로토타입 게이트웨이 데이터 처리

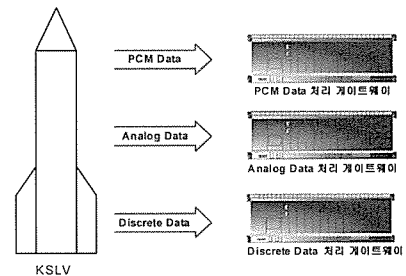


그림 8 다중의 게이트웨이를 통한 데이터 처리 향상 방안

따라서 실제 발사체의 데이터를 처리 하는 방식인 다중의 게이트웨이를 설치하여 각각의 데이터를 처리하면 데이터의 전송속도를 향상시킬 수 있다. 또한, 명령 제어 서버와 데이터 분배 서버의 경우도 하나의 워크스테이션에서 논리적인 두 가지 서버의 기능을 수행하여 데이터 처리에 부하가 많이 걸렸으므로 성능향상을 위해서는 각각의 서버를 물리적으로 분리 구성하여야 할 것이다. 측정 End Item의 현재 값을 관리하는 CVT(Current Value Table)는 공유 메모리나 MMDB(Main Memory Database)를 사용하여 분리된 명령 제어 서버나 데이터 분배 서버가 접근 가능하도록 구현하면 보다 향상된 성능을 구현할 수 있다.

소프트웨어적인 측면에서 현재 데이터 분배 서버의 구조(아키텍처)는 한정된 시스템 자원으로 SSR(10ms) 내에 모든 데이터 처리를 해야 하는 제약조건으로 인해 스레드 기반의 아키텍처로 설계, 구현되어 있다.

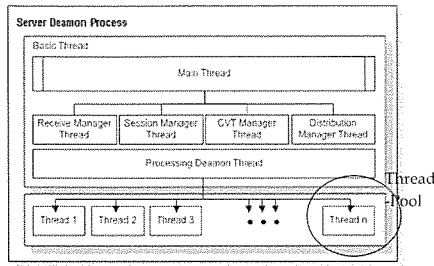


그림 9 데이터 분배 서버 스레드-풀 구성

하지만 그 구현과 검증이 어렵고 너무 많은 작업 스레드를 사용하다 보면 시스템의 자원을 효율적으로 이용할 수 없는 경우도 발생할 수 있으므로 데이터 수신 및 처리 과정을 스레드 풀 (Thread Pool) 구조로 변경하여 처리 부하를 감소시키면서 시스템 자원을 효율적으로 이용할 수 있도록 구성해야 할 것이다.

6. 결 론

본 논문에서는 발사관제시스템 프로토타입의 실시간 데이터 처리능력을 분석하고 성능 향상에 대해 연구하였다. 단위 테스트 시 실시간 처리 능력을 보았을 때 하드웨어적으로는 버퍼 점유 시간 최소화 및 소프트웨어적으로는 Jitter를 최소화 하여 계산루프의 정확성을 최대화함으로써 실시간 데이터 처리 성능이 우수 하는 것을 확인 할 수 있었다. 그러나 발사관제시스템 프로토타입 전체 테스트 시 PCM 생성과 전송을 하는 시뮬레이터와 데이터 분배 서버에서 시스템의 처리 기능 증가로 인해 처리해야 할 프로세스가 많게 되어 SSR 10ms로 데이터를 처리를 못하는 것을 확인하였다. 이는 하드웨어의 데이터 처리 능력의 한계로 인한 원인과 소프트웨어 구조론 상의 원인이 있으므로 하드웨어 교체를 통한 성능 및 소프트웨어 구조를 스레드-풀 방식으로 구성하면 발사관제시스템 프로토타입 성능 향상을 예상할 수 있다. 이와 같은 발사관제시스템 프로토타입에서 발생한 문제를 해결함으로써 나로우주센터 발사관제시스템 구축 시 위험 요소를 줄일 수 있을 것이다.

7. 참 고 문 헌

- [1] 서진호, 홍일희, 정의승, 정해욱 “발사관제시스템 프로토타입 개발”, 한국항공우주학회 추계학술논문 발표회, 2004. 11. 19, pp1019 ~ 1023
- [2] 최원, 정해욱, 서진호, 홍일희, “KSLV-I 발사관제센터 시뮬레이션 시스템 실시간 데이터 처리를 위한 프로토타입 시험평가”, 한국항공우주학회 춘계학술논문 발표회, 2004. 4. 16, pp603 ~ 608
- [3] 서진호, 신명호, 홍일희, 이영호, “KSLV-I 발사관제시스템 개발개념설계”, 한국항공우주학회 추계학술논문발표회, 2003. 11. 14, pp1187 ~ 1190
- [4] 서진호, 홍일희. “우주발사체 발사시험을 위한 실시간 원격제어시스템 개념 설계”, 제 4회 우주 발사체 기술 심포지움, 2003. 5. 3, pp344 ~ 347
- [5] W.E. Hammond, "Design Methodologies for space Transportation Systems", Reston, AIAA Inc., 2001.

[6] 광두영, “LabVIEW Express”, Ohm사, 2003.

[7] Kirk Loufheed, Wayne Prince, "Checkout and Launch Control System(CLCS) System Design Document(SDD), Volume 3A, Section 6. through 6.7. CLCS System Design Topics", NASA CLCS System Engineering Office, 2002. 2.4

[8] 이상래, 이수진, 이재득, “RS-485 버스를 이용한 KSR-III 데이터 취득시스템 개발”, 한국항공우주학회지, Vol 29, 2001.10, pp103 ~ 110.

[9] 소형위성발사체(KSLV-I) 개발사업(I), 한국항공우주연구원, 2003.7.16

[10] <http://www.sbs.com/products/341>