



J. Korean Soc. Aeronaut. Space Sci. 49(6), 481-488(2021)

DOI:https://doi.org/10.5139/JKSAS.2021.49.6.481

ISSN 1225-1348(print), 2287-6871(online)

국내 개발 인공위성을 위한 저비용 원격명령 처리 시스템 구현 및 분석

박상섭¹, 이성진², 전용기³

Development and Analysis of Low Cost Telecommand Processing System for Domestic Development Satellites

Sang-Seob Park¹, Seongjin Lee² and Yong-Kee Jun³Department of Aerospace Engineering, Gyeongsang National University¹Department of AI Convergence Engineering, Gyeongsang National University²Department of Aerospace and Software Engineering, Gyeongsang National University³

ABSTRACT

The satellite telecommand processing system is the only way to provide telecommands for status monitoring, control, and mission execution. Domestic satellites can be divided into science, technology, and multi-purpose satellites, and geostationary satellites. These Satellites uses CCSDS standard protocol to communicate with ground stations. However, existing domestic satellites use only software to decode telecommands which increases cost of software development and verification of the developed software. Performance of software only approach is relatively low compared to hardware. In this paper, we present ASIC processing system specifically designed to decode telecommands. The system consists of a telecommand RAM, a protocol RAM/ROM, an ASIC, an interface unit of FPGA, and a relay block. The system handles general commands and pulse commands that are used in satellites. We established a ground station equipment and test environment to verify the system functionality, The result shows that our system reduces the development cost by 1/5 and improves the performance by 10^5 times compared to the previous systems that decode telecommands only by software.

초 록

인공위성의 원격명령 처리시스템은 상태감시, 제어 및 미션 실행을 위한 원격명령을 제공하는 유일한 통로이다. 국내의 인공위성은 과학 및 기술 위성, 다목적 위성 및 정지궤도 위성으로 나눌 수 있으며 CCSDS 표준 프로토콜을 사용하여 지상국과 통신을 수행한다. 그러나 기존의 국내개발 위성은 소프트웨어를 사용하여 원격명령어를 디코딩하여 소프트웨어 개발 및 검증 비용이 높고 하드웨어와 비교할 때 상대적으로 성능이 낮다. 본 연구에서는 원격명령 디코딩 ASIC을 이용한 원격명령 처리시스템을 제시한다. 이 시스템의 하드웨어는 telecommand RAM, protocol RAM/ROM, telecommand ASIC, interface FPGA 및 relay block으로 구성되었다. 이 시스템은 인공위성이 사용하는 일반 명령 및 펄스 명령을 처리한다. 시스템을 시험 및 검증하기 위해 점검 장비 및 시험환경을 구축하였다. 제안한 ASIC 기반의 telecommand 처리시스템은 소프트웨어 기반 디코딩 시스템에 비해 개발 비용을 1/5로 줄였을 뿐만 아니라 성능은 10^5 배 향상되었다.

Key Words : Satellite(인공위성), Telecommand(원격명령), CCSDS Protocol

† Received : January 29, 2020 Revised : April 13, 2021 Accepted : April 16, 2021

¹ Senior Research Engineer, ² Associate Professor, ³ Professor

³ Corresponding author, E-mail : jun@gnu.ac.kr

© 2021 The Korean Society for Aeronautical and Space Sciences

I. 서 론

인공위성은 지구의 중력에 의해 지구 주위를 공전하는 인공적인 천체를 의미한다. 인공위성의 원격명령 시스템은 위성을 감시, 제어 및 임무 수행을 위한 원격명령어를 제공하는 유일한 통로이다. 우리나라의 인공위성은 사용자의 요구에 따라 크게 과학기술위성, 다목적실용위성 및 정지궤도위성으로 나눌 수 있다.

지금까지 국내에서 개발해온 인공위성은 지상국과 CCSDS(Consultative Committee for Space Data System)[1-4] 표준 프로토콜을 사용하여 통신하는데, CCSDS 프로토콜 계층의 상위 2개 계층은 소프트웨어를 이용하여 디코딩한다. 이로 인하여 소프트웨어의 개발 및 검증 비용이 많이 소요되고 하드웨어 대비하여 상대적으로 수행성능이 낮다. 또한 위성 컴퓨터의 작업이 가중되어 시스템의 처리 속도가 느려질 우려가 있다.

이러한 문제점을 공유하고 있는 선진 인공위성 제작기관들도 원격명령 시스템의 개발 비용과 기간을 줄이기 위한 노력을 기울이고 있으며, 나아가 인공위성과 지상 관제국과의 통신을 위한 기존 인프라의 재활용도 적극 추진하고 있다.

본 연구에서는 원격명령 시스템의 이러한 비용 및 성능 문제를 해결하기 위해서 디코딩 ASIC을 적용한 시스템을 제시한다. 제시된 시스템의 하드웨어는 크게 telecommand RAM(Random Access Memory), protocol RAM/ROM(Read Only Memory), telecommand ASIC(Application Specific Integrated Circuit), interface FPGA unit 및 relay block으로 구성된다. 입력신호는 트랜스폰더로부터 전달받은 원격명령 직렬 데이터, FPGA 제어를 위한 신호, 서브시스템으로부터 입력받은 설정 신호 및 시스템 동작 필수 신호들로 구성된다. 출력으로는 디코딩한 원격명령어를 포함하여 FPGA에서 접근 가능한 데이터와 서브시스템으로 전달되는 명령어가 포함된다. 또한 인공위성에서 사용하는 명령어인 General Command와 Pulse Command를 처리할 수 있다.

시스템의 시험 및 검증을 위해서 점검장비 및 시험환경을 구축하였다. 시스템의 기능을 확인하기 위하여 기능시험을 수행하였으며, 모든 시험은 먼지나 불순물이 침투하지 못하고 항온, 항습, 항압이 유지되는 clean room에서 수행되었다. 시험은 PC의 시험 소프트웨어를 통하여 제어하며, 점검장비는 원격명령 시스템에 데이터를 요청하고 그 결과를 받아서 시험 소프트웨어에 전달한다.

본 연구에서는 ASIC을 적용한 원격명령 시스템은 소프트웨어로 디코딩을 수행하는 시스템에 비하여 개발 비용의 감소와 성능이 개선됨을 보인다. 디코딩 소프트웨어의 비용은 LOC(Line of Code)를 척도로 사용하고 ASIC의 비용은 하드웨어의 구매비용을 적

용하여 비용 측면의 효과를 분석하였다. 성능 측면은 소프트웨어의 실행시간 측정값과 ASIC에서 지원하는 처리속도를 비교하여 분석하였다.

II. 본 론

2.1 연구배경

2.1.1 연구분야

인공위성 TMTC(telemetry and telecommand) 시스템의 기능은 크게 두 가지로 요약된다. 첫째는 지상국으로부터 원격명령(telecommand)을 수신하여 디코딩한 결과를 인공위성의 모든 장치에 전달하는 기능이다. 둘째는 인공위성의 장치들로부터 상태정보를 전달받아 지상국으로 전송하는 원격데이터측정(telemetry) 기능을 포함한다.

인공위성은 제어, 명령, 통신 및 데이터 반환을 위하여 지상국과의 광범위한 통신이 필요한데, 이를 위한 접속은 원격명령 기능을 통해서만 이루어진다. 원격데이터측정 기능은 원격 또는 접근할 수 없는 지점에서 위성의 상태를 측정, 수집 및 모니터링을 위해 지상국으로 데이터를 전송하는 기능이다.

인공위성 통신 시스템의 기본 구조는 Fig. 1에서 보이는 바와 같이 트랜스폰더가 지상국의 업링크를 통해 원격명령어를 수신하고 무선신호를 복조하여 TMTC 시스템으로 전송한다. 반대로 TMTC 시스템은 원격측정 데이터를 포맷팅하여 트랜스폰더로 전달하여 다운링크를 통하여 지상으로 내려 보낸다.

위성 관련 기관들은 60년대 이후 데이터 구조의 표준을 사용하였으며 데이터 시스템의 복잡성이 증가함에 따라 표준화의 필요성이 점차 증가하게 되었다. 1982년 CCSDS를 설립하기 위한 위원회가 발족되었고 국제 우주 통신 분야에서 폭넓게 사용되고 있으며, ESA(European Space Agency)는 CCSDS의 회원으로서 상당 부분을 CCSDS 규격을 따르고 있다[1-4].

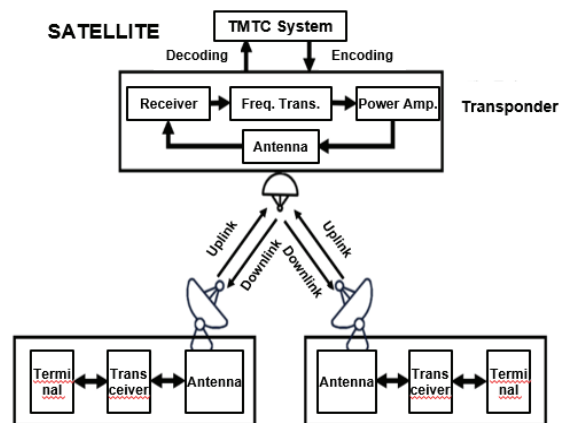


Fig. 1. Basic Composition of Satellite Communication System [5]

CCSDS 표준은 패킷 표준이라고도 불리며 ESA에서는 이 표준을 따라 원격명령과 원격측정에 대하여 PSS (Procedures, Specifications & Standards) 규격으로 제정하여 사용하고 있다.

CCSDS의 원격명령(telecommand; TC) 아키텍처는 매우 다양한 위성의 임무에 적용 가능하도록 계층화되고 표준화된 명령 서비스의 복잡한 집합의 형태로 되어 있다[6]. Fig. 2와 같이 총 7개의 독립적인 계층으로 구성되어 있으며 각 계층은 상위 계층의 특정 서비스를 제공한다. CCSDS 계층은 크게 TC Data Management Service, TC Data Routing Service, TC Channel Service로 나눌 수 있다. Data Management Service는 Application Process Layer, System Management Layer, Packetization Layer를 포함하며, 정의된 사용자 사이로의 명령을 전송하고 전체 원격명령 시스템의 조화 및 관리의 기능을 수행한다. TC Data Routing Service는 Segmentation Layer와 Transfer Layer를 포함하며, 채널의 다중화 기능을 제공하고 원격명령 데이터를 전송할 때 에러 제어를 통하여 신뢰성을 보장한다. TC Channel Service는 Coding Layer와 Physical Layer를 포함하며, 지상으로부터 인공위성으로의 신뢰성 있는 단일 통신 경로를 제공하며 에러 제어 기능을 지원한다.

국제적으로 원격명령시스템의 제어 로직의 개발 추세는 크게 FPGA를 이용한 로직 설계와 ASIC을 통한 기능 구현에 있다[7-12]. 두 경우 어느 곳이나 IP(Intellectual Property) 코어 형태로 설계와 합성이 가능하다. 원격 명령은 지상에서 원격에 위치한 물리적인 장치인 위성으로 정보를 전달하며, 위성의 다양한 조작에 필요한 다양한 유형의 명령을 생성해야 한다[7]. 개별 플랫폼의 요구사항을 바탕으로 직접 모든 로직을 설계하여 로직의 유연성이나 추후 확장에 대비하여 FPGA를 설계하는 것은 소수 위성을 위한 맞춤형 설계에 적당하다. 특히 여러 기능을 하나의 FPGA에 통합하여 특정 위성을 위한 설계를 구현할 수 있다는 이점이 있다.

SoC(System on Chip) 설계가 갖는 장점은 고성능, 소형화, 낮은 배터리 소모율 및 비용 절감 등이 있다[8]. 또한 과거의 여러 개의 칩을 독립적으로 활용하던 방식과 달리 하나의 칩으로 여러 기능을 통합하기 때문에 개발 복잡도를 크게 줄일 수 있다. 그러나 하나의 칩으로 모든 것을 통합하기 위해서는 또 다른 수준의 복잡도를 만들기 때문에 기능 단위의 통합을 위한 새로운 방법론과 기법이 개발되었다. 자체 개발하거나 타사에서 얻은 IP 코어(가상 구성 요소)라고 불리는 사전 설계된 블록을 SoC로 설계하여 단일 칩으로 결합하는 방식이다. 이와 같이 ASIC(Application Specific Integrated Circuit)을 적용한 SoC 설계는 기존의 IP 코어를 재사용하기 때문에 특정 기능을 적용하고 구현하는 있어 개발 기간 낮추고 생산성을 높이는 데 있어 매우 효과적이다.

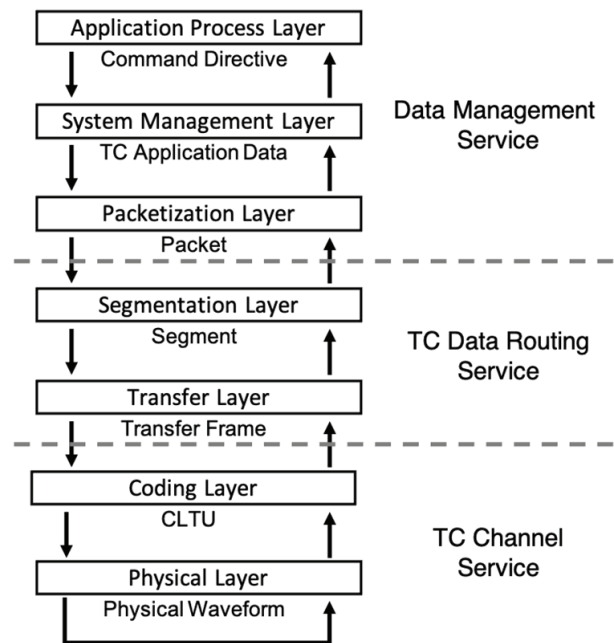


Fig. 2. CCSDS Telecommand Protocol Architecture [6]

2.1.2 문제제기

기존의 국내 인공위성은 통신 프로토콜의 국제 표준 규격인 CCSDS 규격을 적용하여 원격명령 시스템을 구현하였다. 하지만 원격명령 CCSDS 프로토콜의 상위 계층을 소프트웨어만으로 디코딩하는 문제가 있으며, CCSDS 표준 프로토콜의 명령어 디코딩 관련된 4개 계층 중에서 상위 2개 계층을 소프트웨어로 디코딩하는 것을 알 수 있다[13-15].

이와 같이 인공위성 원격시스템에서 원격명령 프로토콜 상위 2개 계층을 소프트웨어로 디코딩함으로써 제어컴퓨터의 소프트웨어 개발 및 검증 비용과 수행성능의 손실이 발생하게 된다. 첫째, 소프트웨어를 개발하기 위하여 투입되는 인력과 시간은 비용의 상승으로 연결되며 이는 최근의 저비용의 효율적인 인공위성을 개발하려는 세계적인 추세와 어울리지 않는다. 특히 안전이 중요한 시스템에 사용되는 소프트웨어는 오늘날 시스템 설계의 주요 구성 요소가 되었으며, 항공기나 인공위성의 안전을 보장하기 위하여 소프트웨어의 높은 신뢰성이 요구된다[16]. 항공기 소프트웨어의 지침을 수립하고 수명주기를 다루는 DO-178을 인증함으로써 제품의 신뢰성, 유지관리성을 보장할 수 있다. 산업 조사에 따르면 DO-178B 인증을 위하여 유사한 비인증 상용 프로젝트와 비교하여 75%~150%의 추가 비용이 발생한다[16]. DO-178B 결과를 고려하여 분석한 결과를 Table 1에 정리하였으며, 이 수치는 DO-178B의 결과이며 추가적인 검증목표로 인하여 DO-178C에서 노력과 비용을 증가시킨다.

Table 1. Composition of DO-178B Process [16]

DO-178B Process	Ratio (%)
Planning (Preparing for the planning documents)	16
Planning (Project management)	11
Software Development	41
Verification	23
Integral Process (Quality assurance)	5
Integral Process (Configuration management)	4

둘째, 소프트웨어의 작업량이 가중되어 시스템 처리 속도가 저하될 수 있다. 원격명령 디코딩을 소프트웨어가 수행함으로써 인공위성의 임무 운용이라는 본연의 업무 이외에 원격명령어 디코딩 작업이 가중되어 수행성능의 부족으로 이어진다. 소프트웨어로 운용되는 작업들 중 원격명령어 디코딩 업무를 수행하는 명령 프로세스를 제외시킴으로써 시스템의 전체적인 처리 속도를 높이는 데 기여할 수 있다[17,18].

최근 미국과 유럽의 위성 제작기관들은 위성체와 지상 관제국과의 통신장치인 원격명령 시스템의 개발 비용과 기간을 줄이기 위한 노력을 기울이고 있다. 현대 인공위성의 컴퓨터 및 서비스시스템의 기능이 향상됨에 따라 처리량에 대한 요구가 증가하고 있으며, 이는 더욱 강화된 재정적인 제약과 함께 더 나은 기술적인 환경으로 인하여 비용 절감 및 원격 명령의 기능과 효율성을 높이는 공통적인 요구사항으로 이어지고 있다[17]. 또한 저비용의 소형 위성이 기존의 중형 이상이 담당하는 임무를 대체하고 있는 추세이다. 더욱이 안전에 중요한 시스템을 위한 인증가능한 소프트웨어 개발은 시간과 노동 집약적이며, 따라서 필요한 비용 및 노력을 최소화하기 위하여 효과적이고 효율적인 기술이 필요하다[16]. 추가적으로 소프트웨어의 복잡도가 증가할지라도 추가적인 소프트웨어의 요구 없이 본래의 하드웨어의 기능만으로 구현이 가능하여야 할 것이다. 본 연구에서는 원격명령 디코딩 ASIC(Application Specific Integrated Circuit)을 사용하는 시스템을 제안하여 국내 개발 인공위성을 위한 저비용 원격명령 시스템 개발 방안을 소개한다.

2.2 시스템 개발

2.2.1 설계

Figure 3은 원격명령시스템의 하드웨어 구성을 나타낸다. TC RAM, Protocol RAM과 ROM, TC ASIC, Interface FPGA 및 AU Relay 블록으로 구성되어 있

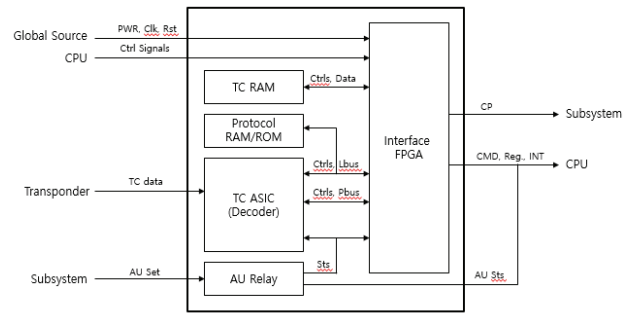


Fig. 3. Hardware Configuration of Telecommand System

다. 입력은 트랜스폰더로부터 전달받은 직렬의 원격명령 데이터, CPU로부터 입력되는 FPGA 제호 신호, 서브시스템으로부터 입력되는 설정 신호, 시스템 활성화를 위한 Global Source로 구성되어 있다. 출력은 CPU로 전달하는 FPGA가 접근 가능한 데이터 및 상태 정보들, 서브시스템에 전달하는 하드웨어 단독실행 명령어로 구성되어 있다. 화살표의 방향은 신호의 이동 방향을 나타내며, 여러 블록에서 공유하는 경우 화살표를 분기시켜서 표현하였다.

원격명령어를 디코딩하는 TC ASIC 블록을 중심으로 Protocol ROM은 Fig. 4에서 볼 수 있듯이 명령어를 디코딩하는 과정 중에 데이터가 올바른지(Valid) 검증하는데 필요한 Frame Header와 Packet Header 정보가 포함되어 있으며, Protocol RAM은 Buffer로 할당되어 디코딩이 이루어지는 저장소를 제공한다. TC RAM 블록은 Interface FPGA 블록에서 사용하는 SRAM이며, 디코딩이 완료된 데이터를 저장하는 용도로 사용된다. Interface FPGA 블록은 TC ASIC 블록으로부터 수신한 디코딩한 TC 데이터를 SRAM에 저장하고, CPU와 버스 통신을 통하여 디코딩한 TC 데이터, 메모리 및 레지스터 값, 디코딩 완료 인터럽트 및 시스템 상태를 알려준다. 또한 AU Relay 블록으로부터 상태정보를 받아서 레지스터에 저장한다. AU Relay 블록은 서브시스템으로 설정 신호를 받아 구동되며, Interface FPGA 블록 및 CPU로 상태정보를 알려준다.

원격명령시스템은 일반적인 상황에서의 명령어인 General Command와 하드웨어 직접실행 명령어인 Pulse Command로 나누어진다. Fig. 5는 General Command의 생성 및 동작 순서를 나타낸다. 화살표는 General Command의 입력에서부터 출력까지의 경로를 나타내며, 화살표 안의 숫자는 동작 순서를 표시한다. 트랜스폰더로부터 General Command가 포함된 TC 데이터가 입력된다. 데이터는 클럭, 데이터, Presence로 구성된다. TC ASIC에서는 AU Relay 블록으로부터 전송된 인증모드 상태와 Protocol RAM과 ROM을 이용하여 디코딩을 수행한다. 디코딩이 끝나면 완료 신호를 발생시키고 Interface FPGA로 데

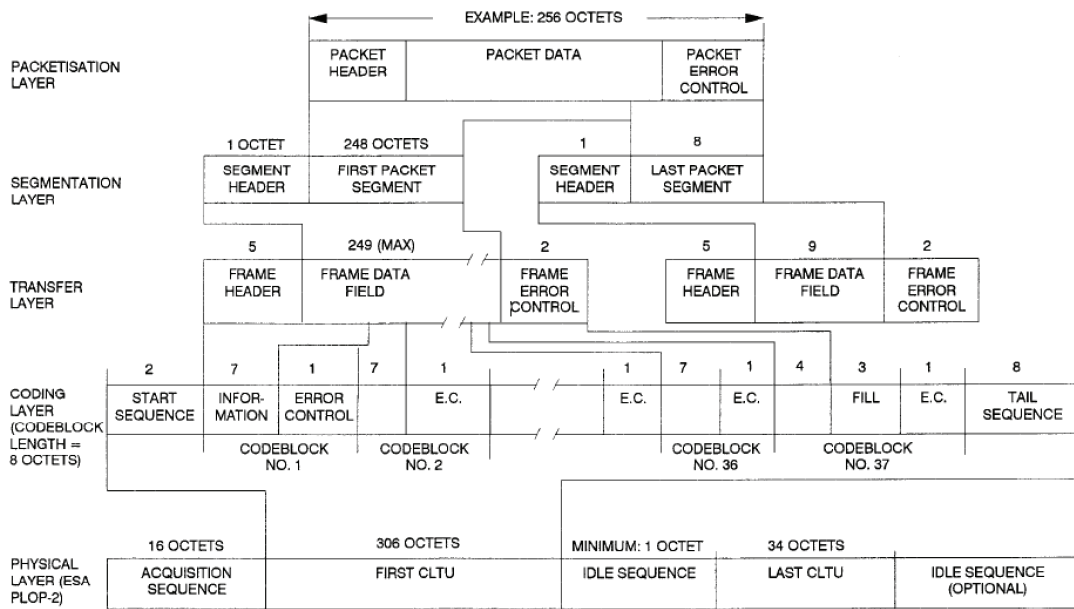


Fig. 4. Example of the Successive Data Structures Generated by the Lower Layers of the Telecommand System [6]

이터를 전달한다. Interface FPGA는 수신된 원격명령 데이터를 TC RAM에 저장한 후 CPU로 인터럽트를 발생한다. CPU가 요청하는 경우 TC RAM에 저장된 General Command를 읽어들이어 전송한다.

Figure 6은 Pulse Command의 생성 및 동작 순서를 나타낸다. 화살표는 Pulse Command의 입력에서부터 출력까지의 경로를 나타내며, 화살표 안에 숫자는 동작 순서를 표시한다. 트랜스폰더로부터 Pulse Command가 포함된 TC 데이터가 입력된다. 데이터는 클럭, 데이터, Presence로 구성된다. TC ASIC에서는 AU Relay 블록으로부터 전송된 인증모드 상태와 Protocol RAM과 ROM을 이용하여 디코딩을 수행한다. 디코딩이 끝나면 완료 신호를 발생시키고 Interface FPGA로 데이터를 전달한다. Interface FPGA는 수신된 원격명령 데이터의 채널 값에 따라 각 서브시스템으로 펄스를 직접 발생시킨다. 이 과정은 소프트웨어의 간섭이 없이 하드웨어 단독으로 실행이 된다.

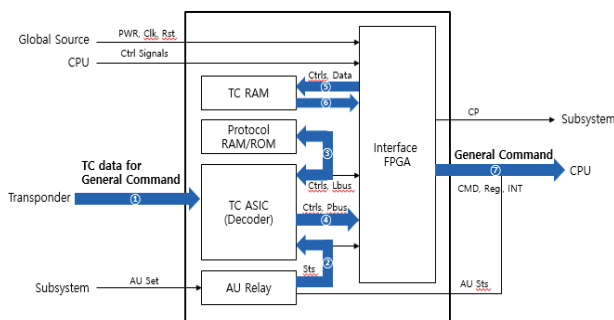


Fig. 5. General Command Creation and Operation Sequence

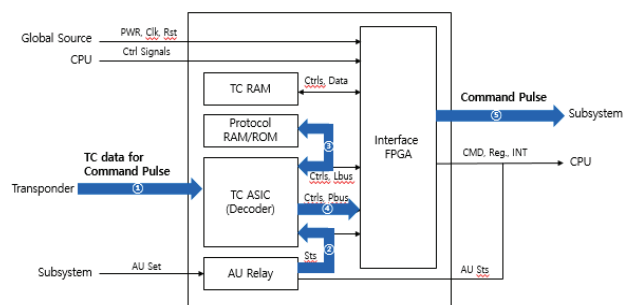


Fig. 6. Pulse Command Creation and Operation Sequence

2.2.2 구현 및 검증

블록 간 혹은 외부와의 데이터 이동을 위한 인터페이스 구성은 크게 세 가지로 구분된다. TC ASIC과 Protocol RAM/ROM 그리고 Interface FPGA와 Protocol RAM/ROM은 버스를 공유하고 있으며 8bit의 데이터 버스를 사용하고 있다. TC ASIC과 Interface FPGA 사이의 데이터 이동은 16bit 데이터 버스를 사용하고 있으며 Interface FPGA와 CPU는 32bit 버스를 사용하여 통신을 수행하고 있다. AU Relay의 경우 두 가지 상태를 출력할 수 있으며 두 개의 출력핀을 통하여 같은 상태 정보를 동시에 내보내게 된다. Fig. 7은 원격명령 시스템의 기능 블록별로 입출력 신호들을 나타낸다.

하드웨어를 검증하기 위한 시험 소프트웨어를 직접 개발하였으며 개념적인 구성은 Fig. 8과 같다. Boot ROM Software, Internal Test Software, EGSE Test Control & Monitor Software로 구성된다. Boot ROM

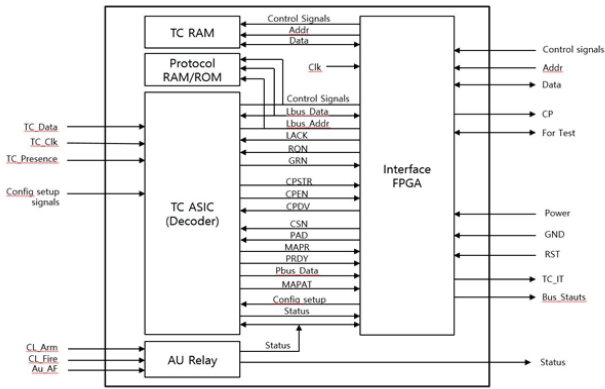


Fig. 7. Block Diagram of Telecommand System

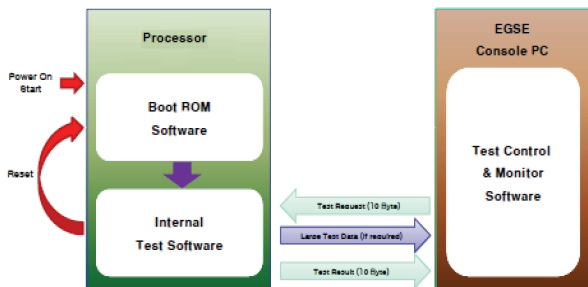


Fig. 8. Test Software Configuration

Software는 Boot ROM에서 동작하는 프로그램으로써 Boot Loader와 Monitor Software로 구성된다. Boot Loader는 전원이 인가되거나 Reset 되었을 때 실행되며, Processor를 초기화하고 Monitor Software를 구동시킨다. Monitor Software는 UART 입출력을 통하여 Interpreter 서비스를 제공하며, 시험 프로그램을 CPU RAM 영역으로 Loading하고 구동시킨다. Internal Test Software는 제어컴퓨터 내부에서 동작하며 EGSE Test Control & Monitor Software로부터 Test Request Command를 수신 후 시험 항목별로 정의된 순서에 따라 시험을 수행한다. 시험이 끝나면 시험 결과를 UART 통신을 통해 EGSE Test Control & Monitor Software로 전송한다. Test Control & Monitor Software는 시험용 PC에서 동작하는 프로그램으로써 EGSE와 범용 계측장비를 제어한다. EGSE와 유닛의 상태를 모니터링하며 시험 시작을 알리는 Test Request Command를 전송하고, 수신한 시험결과를 분석하여 Display 한다.

2.3 실험 및 고찰

2.3.1 실험 설계

본 연구에서 제안하는 시스템에 대한 실험은 크게 비용과 성능 측면으로 진행하였다. 원격명령어 디코딩 소프트웨어와 디코딩 ASIC의 비교 분석을 위한 한계가 존재한다. 원격명령어 시스템이 디코딩을 수행하므로 별도의 디코딩 소프트웨어를 개발하지 않았

다. 따라서 시험용 PC의 시험 소프트웨어에 구현되어 있는 시험용 소프트웨어의 원격명령어 인코딩 프로그램의 코드를 기준으로 디코딩 소프트웨어의 비용을 산정하기로 한다. 예상 비용은 LOC(Line of Code)를 척도로 사용하여 산정하였다. 본 논문에서 사용된 인코딩 프로그램의 소스코드는 약 1,000줄이다. 각 계층의 syntax와 프로토콜에 맞게 인코딩 코드가 작성되어야 하기 때문에 Coding Layer, Segmentation Layer 그리고 Packetization Layer의 3개 계층을 인코딩할 경우 소스코드의 크기는 최소 3,000줄이 된다. 인코딩 소프트웨어의 실행시간은 한 비트 처리 기준으로 Clock 함수를 이용하여 측정하였으며, 반복적으로 수행한 결과의 평균값을 적용하였다. 디코딩 ASIC의 경우 하드웨어 구현에 적용된 4 Mhz 클럭을 기준으로 계산하였다.

2.3.2 결과 고찰

디코딩 소프트웨어의 비용과 원격명령어 시스템에서 사용하는 디코딩 ASIC의 비용을 비교하여 Table 2에 정리하였다. 소프트웨어는 개발비용과 검증비용을 합산한 금액이며, ASIC의 비용은 하드웨어의 구매비용으로 나타내었다. 소프트웨어로 telecommand를 디코딩하는 프로그램을 개발할 경우 약 27,500만원이 필요할 것으로 예상된다. ASIC의 경우 환경모델과 비행모델의 부품 수량을 각각 2개씩 적용하는 경우 약 5,280만원이 소요된다.

Table 3은 디코딩 소프트웨어 비용 산정 항목을 나타낸다. 인공위성의 개발 목표와 사용 장치의 스펙은 상이하기 때문에 CCSDS 프로토콜 처리 기능은 직접 코딩되거나 SOC로 개발되어야 한다. 이때 월 평균 생산코드 500라인을 기준으로 6M/M (Man/Month)가 소요된다고 하고 요구도 분석 기간은 12M/M로 산정하였다. 그리고 GUI 설계 기간 2M/M와 납품 후 유지 및 보수 기간 2M/M가 소요되는 것으로 예상하였다. 또한 프로젝트의 난이도를 고려하여 DO-178B 기

Table 2. Cost Comparison Between Software and ASIC

	Software	ASIC
Contents	Development Cost Verification Cost	Purchase Cost
Cost	Development : ₩110,000,000 Verification : ₩165,000,000 Total : ₩275,000,000	Flight Model : \$17,000*2=\$34,000 Qual. Model : \$7,000*2=\$14,000 Total : ₩52,800,000

준으로 소요되는 검증 비용은 개발 비용 대비 약 150% 수준으로 책정하여 소프트웨어 개발 비용은 총 27,500 만원으로 산정되었다[16]. 여기에는 DO-178C의 추가 검증목표로 인한 비용, STB(Software Test Bed) 시험과 FSW(Flight Software)와의 통합 비용 등을 제외하여 보수적으로 산정하였다. 디코딩 ASIC 적용에 따른 주변 하드웨어 인터페이스 비용은 위의 STB 시험 및 FSW 통합비용과 비교하여 낮은 수준이며 따라서 비용 비교 항목에서 제외하였다. 비용 분석을 통하여 원격명령 시스템에서 디코딩 ASIC을 사용하여 소프트웨어의 개발 및 검증 비용을 생략함으로써 비용을 획기적으로 절감할 수 있음을 확인하였다.

원격명령 시스템의 디코딩 기능을 수행하는 소프트웨어의 성능과 TC ASIC의 성능을 Table 4에 정리하였다. 개발한 소프트웨어의 실행시간은 한 비트 처리 기준으로 Clock 함수를 이용하여 측정하였으며, 총 100회 수행하여 평균값을 적용하였다. 약 50,000 DMIPS 성능을 나타내는 컴퓨터에서 소프트웨어를 실행하였을 경우 실행시간이 0.65ms로 측정되었다.

인공위성 제어컴퓨터의 CPU가 약 15 DMIPS 수준인 경우를 반영하면 2.2s의 실행시간이 소요될 것으로 예상된다. 인코딩/디코딩 소프트웨어는 여러 계층(Transfer Layer ↔ Coding Layer, Segment Layer ↔ Transfer Layer) 간의 데이터를 인코딩/디코딩해야 한다. 또한, Packetization Layer와 Transfer Layer 사이의 데이터를 syntax에 맞게 포맷팅하는 기능도 구현되어야 한다. 이를 모두 구현할 경우 최소 세 배 이상의 실행시간이 소요될 것이다. 디코딩 ASIC이 동작하는 최대 4Mhz 클럭으로 동작하며 최대 50 kbps를 지원한다. 이 경우 하드웨어 내부의 1비트 처리 시

Table 4. Performance Comparison Between Decoding Software and ASIC

Item	Software	ASIC
Measurement Base	Satellite CPU performance : Approximately 15 DMIPS	Max. clock frequency : 4Mhz
Decoding Time	One bit processing time : 2.2s	One bit processing time : 20us

간은 20us가 소요된다. 따라서 본 연구에서는 원격명령 시스템에서 디코딩 ASIC을 사용함으로써 소프트웨어와 비교하여 우수한 성능을 보임을 검증하였다.

III. 결 론

본 연구에서 디코딩 ASIC을 적용한 저비용 원격명령 시스템을 제안하였다. 지상모델에서 비행모델로 개발이 진행됨에 따라 개발 비용 및 검증 비용이 기하급수적으로 증가하게 된다. 본 연구에서는 검증이 완료된 하드웨어를 사용함으로써 디코딩 소프트웨어의 개발 및 검증을 생략하여 비용이 1/5 수준으로 감소함을 확인하였다. 그리고 ASIC을 적용함에 따라 소프트웨어에 비해 상대적으로 수행성능이 10⁵배 향상되는 것을 확인하였다. 이를 통하여 인공위성 제어 컴퓨터의 작업이 가중되어 시스템 처리 속도가 느려지는 단점을 보완할 수 있을 것으로 예상된다. 또한

Table 3. Estimated Cost of Software Development

Item	Value	Unit	Description	
LOC (Line of Code)	3,000	Lines		
Effort	Requirement Analysis	12	M/M	Project Planning, Management
	Coding	6	M/M	3,000/500 (LOC)/(Monthly average production Code Line)
	GUI Design	2	M/M	GUI Design leadtime
	Maintenance	2	M/M	Maintenance and repair after delivery
Development cost	11,000	Ten thousand won	22*500 (M/M)*(Monthly labor cost per person)	
Verification cost	16,500	Ten thousand won	11,000*1.5 Incurred 75%~150% additional cost compared to non-certified commercial projects based on DO-178B. [16] 150% is applied considering the difficulty of the project.	
Total	27,500	Ten thousand won	(Development cost) + (Verification cost)	

세계적인 개발 추세에 맞추어 최첨단 기술을 확보할 수 있는 계기를 마련하였다.

향후에는 원격명령 시스템과 원격측정 시스템의 기능을 통합시키고자 한다. 단일 칩 설계를 사용함으로써 설계의 유연성을 확보할 수 있고 검증이 완료된 하드웨어를 사용하여 개발 비용과 기간을 획기적으로 절감할 수 있다. 또한 여러 플랫폼에 공통적으로 적용할 수 있는 시스템을 개발하여 이식성을 확보할 수 있을 것이다.

후 기

이 성과는 2021년도 정부(교육부) 재원으로 한국연구재단의 지원을 받아 수행된 기초연구사업(No. NRF-2018R1D1A3B07041838과 No. 2019R1G1A110 0455)입니다.

References

- 1) CCSDS, 200.0-G-6, *Telecommand Summary of Concept and Rationale*, CCSDS, 1987.
- 2) CCSDS, 201.0-B-3, *Telecommand Part 1: Channel Service, Architectural Specification*, CCSDS, 2000.
- 3) CCSDS, 202.0-B-3, *Telecommand Part 2: Data Routing Service, Architectural Specification*, CCSDS, 2001.
- 4) CCSDS, 203.0-B-2, *Telecommand Part 3: Data Management Service, Architectural Definition*, CCSDS, 2001.
- 5) Korea Defense Industry Association, "Configuration of satellite communication system," *Defense & Technology*, 1999, pp. 82~83.
- 6) ESA, PSS-04-107, *Packet Telecommand Standard*, ESA, 1992.
- 7) Vijayalakshmi, K., Lakshmi, C. R. and Tejaswini, M. S., "Design and Implementation of CCSDS Standards Based Telecommand System for Satellites using VHDL," *IEEE International Conference on Recent Trends in Electronics, Information & Communication Technology (RTEICT)*, 2016.
- 8) Rajesvari, R., Manoj, G., Ponrani, M. A. and Joy, M. A., "IP Core Based Architecture of Telecommand System-on-Chip (SoC) for Spacecraft Applications," *IEEE International Conference on Signal Processing, Image Processing and Pattern Recognition (ICSIPR)*, 2013.
- 9) Rajesvari, R., Manoj, G. and Ponrani, M. A., "System-on-Chip(SoC) for Telecommand System Design," *International Journal of Advanced Research in Computer and Communication Engineering (IJARCCE)*, 2013, pp. 1580~1585.
- 10) Kushal, M., Bhanu, P. N. and Shrihari, M. R., "Cost Effective Telemetry and Telecommand System for Small Satellite," *International Journal Of Engineering Research & Technology (IJERT)*, 2017, pp. 286~290.
- 11) Amogh, K. M., Mohan, S., Handral, R. V., Neema, N. and Agarwal, P., "FPGA Implementation of Telecommand System for Nano-satellite," *IEEE 3rd International Conference on Communication and Electronics Systems (ICCES)*, 2018.
- 12) Arunkumar, S. and Kalaivani, T., "FPGA Implementation of CCSDS BCH (63, 56) for Satellite Communication," *IEEE International Conference on Electronics Design, Systems and Applications (ICEDSA)*, 2012.
- 13) Kim, S. Y., Lee, J. J., Gu, B. J., Oh, S. H., Kang, K. I. and Chae, J. S., "Flight Model Design and Implementation of NextSat-1 TMTC (Telemetry and Tele-command) Module," *Proceeding of The Korean Society for Aeronautical And Space Sciences Fall Conference*, November 2017, pp. 970~971.
- 14) Ryu, J. H., Park, J. Y., Jung, S. G. and Lee, H. G., "Development of FPGA of CCSDS Telemetry Telecommand for Small Satellite Systems," *Proceeding of The Korean Society For Aeronautical And Space Sciences Fall Conference*, November 2008, pp. 1377~1380.
- 15) Choi, E. J., Park, J. Y., Jung, S. G. and Lee, H. G., "Development of Software And Design of FPGA for the CCSDS-Based Communication System of A Small satellite," *Proceeding of The Korean Society For Aeronautical And Space Sciences Spring Conference*, April 2006, pp. 956~959.
- 16) Bingol, M. K., Deniz, E., Sari, M., Saritas, I. E. and Yilmazer, Y., "Adapting DO-178C Processes by Implementing a Reverse Engineering Technique," *IEEE/AIAA 33rd Digital Avionics Systems (DASC) Conference*, 2014.
- 17) Epperly, M. E., Walls, B. J. and Wasiewicz, M., "FPGA CCSDS Command Decoder with BCH EDAC and Level-0 Command Execution," *IEEE Aerospace Conference 4*, 2002.
- 18) Kim, S. Y., Lee, J. J., Gu, B. J., Oh, S. H. and Chae, J. S., "Flight Model Design and Implementation of NextSat-1 Command and Data handling Subsystem(CDS)," *Proceeding of The Korean Society For Aeronautical And Space Sciences Fall Conference*, November 2016, pp. 1076~1077.