

차세대 저궤도 위성의 PCI 기반의 1553B 통신 소프트웨어 설계

최종욱*, 정희원, 정재엽*, 정희원, 유범수*

Design and Development of PCI-based 1553B Communication Software for Next Generation LEO On-Board Computer

Jong-Wook Choi*, Jae-Yeop Jeong*, Bum-Soo Yoo*

요 약

현재 한국항공우주연구원에서 개발 중인 차세대 저궤도 위성의 탑재컴퓨터는 높은 성능을 위해 LEON2-FT/AT697F 프로세서를 사용하며 SpaceWire, 1553B, DMAUART, CAN Master 등의 다양한 통신을 지원하기 위해 별도의 FPGA 기반의 통신칩이 개발되었다. 프로세서와 통신칩간의 통신은 PCI 버스를 통해서 이루어지며, 탑재소프트웨어에서 직접 PCI 버스를 통해 각종 디바이스를 제어 및 통신을 수행한다. 차세대 탑재컴퓨터에서는 기존 1553B 통신을 위해 사용되었던 VASI IP1553B 컨트롤러 대신 Actel 1553BRM 코어를 사용하며 통신칩의 AMBA 버스상에 연결을 위해 Aeroflex Gaisler에서 개발 된 B1553BRM Wrapper를 사용한다. 본 논문에서는 차세대 저궤도 위성에서의 PCI기반의 1553B 통신 소프트웨어의 구조와 통신방법에 대해서 기술하고 탑재소프트웨어 레벨의 태스크를 통한 1553B 처리방식에 대해서 설명한다. 그리고 시뮬레이터 및 실제 하드웨어에서 테스트 된 결과에 대해 설명한다.

Key Words : Actel, 1553B, FSW(Satellite Flight Software), PCI, AMBA

ABSTRACT

Currently developing the OBC of the next-generation LEO satellite by Korea Aerospace Research Institute adopts the LEON2-FT/AT697F processor to achieve high performance. And various communication devices such as SpaceWire, MIL-STD-1553B, DMAUART and CAN Master are integrated to the separated standard communication FPGAs within the OBC, where they can be controlled by the processor and flight software (FSW) through PCI interface. The Actel 1553BRM IP core is used for the 1553B in the next-generation LEO OBC and the B1553BRM wrapper from Aeroflex Gaisler is used for connecting it to the AMBA bus in FPGA. This paper presents the design and development of PCI-based 1553B communication software, and describes the handling mechanism of 1553B operation in FSW task level. Also it shows the test results on real-hardware and simulator.

I. 서 론

인공위성의 임무가 다양해지고 요구사항이 높아짐에 따라 탑재컴퓨터의 성능향상이 필수적으로 대두되었으며, 인공위성의 다양한 활용을 위해 표준화 설계, 위성 기능 및 임무에 따라 시스템 재구성성이 가능한 모듈화 설계의 필요성이 계속 제기되었다. 한국항공우주연구원에서 개발된 기존 저궤도 위성의 경우 다양한 내외부 인터페이스를 가진 SPARC v7기반의 MCMERC32 프로세서를 사용한 반면 차세대 저궤도 위성에서는 높은 성능을 만족하기 위해 SPARC v8기반의 LEON2-FT/AT697F 프로세서를 사용한다[1][2]. 그리고

SpaceWire, MIL-STD-1553B, DMAUART, CAN Master 등과 같은 다양한 통신 인터페이스를 별도의 FPGA 기반의 통신칩으로 구성하여 AT697F 프로세서와 PCI 버스를 통해 통신 할 수 있도록 개발되었다[3]. 이를 위해 탑재소프트웨어에서는 새로운 AT697F 프로세서 보드를 위한 VxWorks 실시간 운영체제 BSP (Board Support Package) 개발과 함께 PCI 인터페이스 관련 디바이스 드라이버가 개발되었으며, 각 통신칩의 다양한 통신 디바이스를 제어할 수 있는 디바이스 컴포넌트가 개발 되었다[4]. 본 논문에서는 현재 항우연에서 개발 중인 차세대 저궤도 탑재컴퓨터에서 탑재소프트웨어의 PCI 기반의 1553B 통신 소프트웨어 컴포넌트의 설계

*한국항공우주연구원 위성탑재소프트웨어팀 (jwchoi@kari.re.kr, jyjeong@kari.re.kr, bsyoo@kari.re.kr)

접수일자 : 2016년 9월 6일, 최종게재확정일자 : 2016년 9월 22일

및 구조에 대해 기술하며, 탑재소프트웨어 레벨의 태스크를 통한 1553B 처리방식에 대해서 설명한다. 그리고 시뮬레이터 및 실제 하드웨어에서의 테스트 결과에 대해 설명한다.

II. 차세대 저궤도 위성의 탑재컴퓨터

위성에서 사용되는 프로세서는 그림 1처럼 ERC32 프로세서가 기존 탑재컴퓨터의 메인 프로세서로 사용되었으나, 현재는 LEON2/3 기반의 프로세서가 주로 사용되고 있으며 차세대 프로세서로 Quad Core 기반의 GR740 프로세서가 ESA 주관하에 개발되고 있다. 항우연에서 개발된 기존 저궤도 위성에서는 다양한 인터페이스를 내장한 SPARC v7기반의 MCMERC32 프로세서가 사용된 반면 현재 개발 중인 차세대 저궤도 위성 및 중형위성에서는 SPARC v8 기반의 LEON2- FT 프로세서 중 Atmel에서 개발 된 AT697F 프로세서를 사용한다.

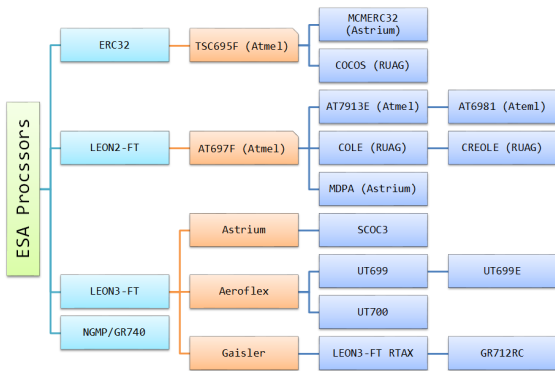


그림 1. Processor in Space

1. LEON2-FT/AT697F 프로세서

기존 ERC32 프로세서의 한계 및 문제점[8]을 극복하기 위해 ESA를 주축으로 개발된 LEON 프로세서는 높은 성능과 함께 다양한 인터페이스를 내장한 SoC (System On Chip) 형태로 개발되었으며, LEON2, LEON3 그리고 LEON4/GR740[9]로 계속 확장되었다.

Atmel에서 개발 된 AT697F 프로세서는 그림 2와 같이 SPARC v8 기반의 Integer Unit (IU), MEIKO Floating-Point Unit (FPU), 32KB 4-way associative Instruction cache, 16KB 2-way associative Data cache, 하드웨어 기반의 multiplier/divider를 가지고 있으며, APB bus에 인터럽트 컨트롤러, 2개의 32bit 타이머, Watchdog Timer (WDT), 2개의 UART, GPIO 등을 가지고 있다. PCI 인터페이스를 위해 PCI/AMBA bridge 기반의 PCI controller를 가지고 있으며, 소프트웨어 로딩 및 디버깅을 위하여 DSU (Debugging Support Unit)과 JTAG를 가지고 있다.

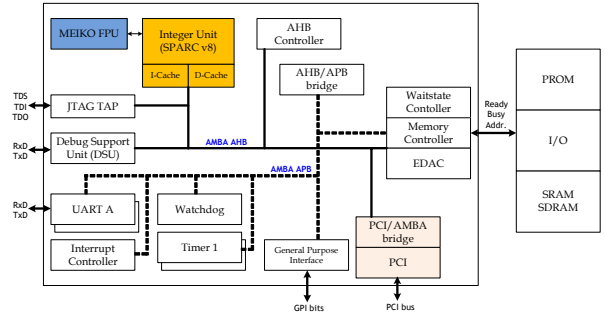


그림 2. AT697F Processor Block Diagram

AT697F 프로세서는 외부 인터페이스를 확장할 수 있도록 PCI를 제공하고 있으며, PCI spec 2.2와 호환되며 33MHz로 동작한다. PCI 인터페이스는 round-robin 알고리즘을 통해 최대 4개의 PCI agent를 관리할 수 있으며, PCI initiator (master) 및 target으로 동작한다. PCI 데이터 전송은 8개의 word로 구성된 4개의 synchronizing data FIFO를 통해서 수행되며, PCI 영역은 AT697F 메모리 맵에서 non-cacheable 영역으로 할당된 0xA000.0000 ~ 0xFFFF.FFFF에 맵핑되어 있으며 PCI 관련 레지스터들은 AMBA 영역 0x8000.0100 ~ 0x8000.0280에 시스템 레지스터로 할당되어 있으며 Host-Bridge 모드로 동작한다.

2. 차세대 저궤도 탑재컴퓨터 프로세서 보드

차세대 저궤도 위성의 탑재컴퓨터 프로세서 보드는 그림 3과 같이 메인 프로세서로 AT697F를 사용하며, 3개의 통신 칩에는 다양한 미션을 수행 할 수 있도록 다수의 통신 코어를 가지고 있으며 AT697F 프로세서와 PCI로 연결되어 있다.

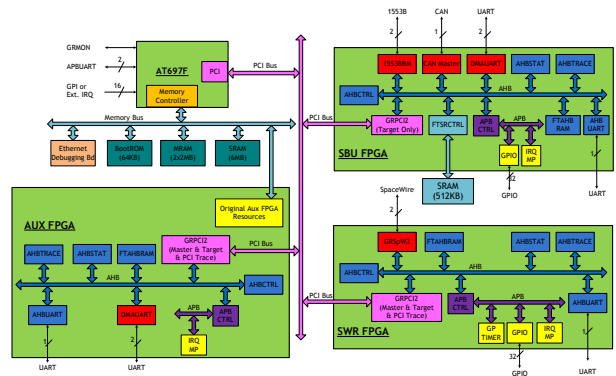


그림 3. Processor Board Architecture

SBU (Serial Bus & UART Control) 통신칩은 GRCPI2, 1553B, CAN Master, DMAUART, FTAHBRAM, IRQMP 및 SRAM 등이 내장되어 있으며, SWR (SpaceWire RMAP) 통신칩은 GRCPI2, GRSpW2, FTAHBRAM, IRQMP 등으로 구성되어 있다. AUX 통신칩은 GRCPI2, DMAUART, FTAHBRAM, IRQMP 등으로 각각 구성되어 있다. 각 통신 칩은 Aeroflex Gaisler의 GRLIB IP 코어[5]와 항우연에서 개

발된 자체 IP 코어를 기반으로 하고 있으며, 각 IP 코어들은 AMBA 버스로 연결되어 있어 새로운 모듈의 추가 및 확장이 용이하다. 이러한 구조는 LEON3/GRLIB 구조[3]와 유사하나 프로세서 모듈 대신 PCI가 AMBA Master로 동작하여 AT697F에서 PCI로 연결되어 각 IP 코어들을 제어 할 수 있다. 각 통신칩에서 발생한 디바이스 인터럽트의 경우 PCI Initiator로 라우팅되며, AT697F 프로세서에서 전용 인터럽트로 할당되어 탑재소프트웨어에서 처리 할 수 있다.

Ⅲ. 탑재소프트웨어 소프트웨어 아키텍처 및 컴포넌트

1. PCI 소프트웨어 컴포넌트

현재 저궤도 위성에서 사용 중인 실시간 운영체제인 VxWorks 5.4 경우 PCI 인터페이스를 지원하지 않기 때문에, AT697F 프로세서의 PCI를 지원하기 위한 VxWorks 5.4 기반의 PCI 디바이스 컴포넌트가 BSP 레벨에서 구현되었으며, 이를 바탕으로 SBU/SWR/AUX 통신칩의 다양한 디바이스를 제어 할 수 있도록 확장되었다.

AT697F 프로세서와 3개의 통신칩 초기화 과정[4]은 크게 AT697F PCI 초기화 과정, 통신칩 PCI 초기화 과정을 통해 이루어지며, 통신칩 초기화 이후 그림 4와 같이 Memory Mapped I/O 방식으로 탑재소프트웨어가 액세스 할 수 있게 된다. 이후 각 통신칩의 디바이스를 위한 초기화 과정과 인터럽트 초기화 과정을 수행한다. 모든 초기화 및 설정이 완료된 이후 탑재소프트웨어에서는 기존 디바이스 드라이버를 제어하는 방식과 동일하게 SpW, CAN, 1553B 등의 다양한 디바이스를 PCI를 통해 제어 할 수 있게 된다.

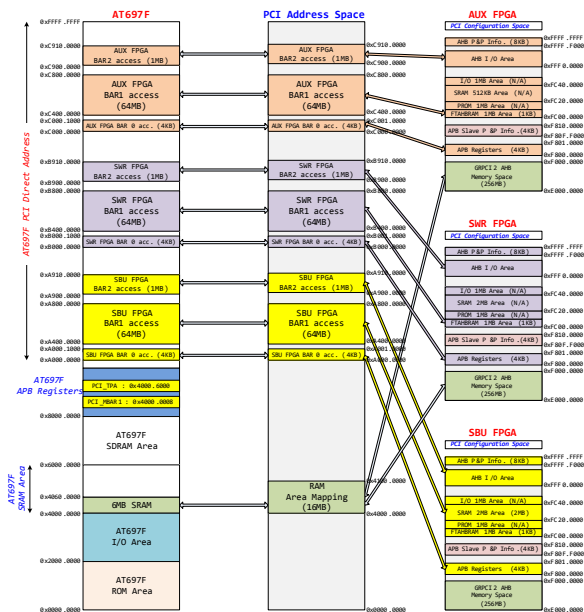


그림 4. AT697F & FPGA PCI Address Mapping

2. 탑재소프트웨어 아키텍처

차세대 저궤도 위성의 탑재소프트웨어는 서비스지향 기반 아키텍처로 설계 및 개발되었으며 그림 5와 같이 6개의 서비스로 구성된다. Core 서비스는 탑재소프트웨어의 태스크를 생성 및 관리, 시스템 초기화 등을 관리하는 Executive 컴포넌트, 위성의 시간을 관리하는 On-Board Time 컴포넌트 그리고 위성 운영시 발생하는 다양한 이벤트와 에러를 리포팅하는 Event/Error 컴포넌트 그리고 Boot ROM S/W 컴포넌트로 구성된다. Health 서비스는 탑재소프트웨어의 상태를 관리 및 모니터링 하는 서비스로, 코드 패치 및 NVMEM Reprogramming을 담당하는 Reprogram 컴포넌트, 주기적인 메모리 스크러빙을 수행하는 Memory Scrub 컴포넌트 그리고 시스템 레벨에서 주요 상태를 모니터링 하는 Monitor 컴포넌트로 구성된다. 지상 명령과 저장 명령을 처리하는 Command 서비스는 TC 컴포넌트와 Stored Command 컴포넌트로 구성되며, 지상으로 원격 텔레메트리를 생성 및 관리하는 Telemetry 서비스는 위성의 다양한 데이터를 획득하는 Data Acquisition 컴포넌트, 실제 원격 텔레메트리를 처리하는 Telemetry 컴포넌트 그리고 대용량 저장 메모리를 관리하는 Mass Memory 컴포넌트로 나뉜다. 응용 서비스는 자세제어를 담당하는 AOCS 컴포넌트, 열제어를 담당하는 TCS 컴포넌트, 전력관리를 담당하는 컴포넌트, 그리고 다수의 탑재체를 관리하는 Payload 컴포넌트로 구성된다. 마지막으로 위성의 다양한 전장품간의 통신을 처리하는 Communication (COMM) 서비스는 TCTM과 SpW 통신을 처리하는 SpW 컴포넌트, IOM과 통신을 처리하는 CAN 컴포넌트, PCDU/VDE와 통신을 수행하는 UART 컴포넌트, 그리고 1553B 통신을 담당하는 1553B 컴포넌트로 구성되며, 모든 통신은 PCI 기반의 통신 디바이스를 통해 수행 된다.

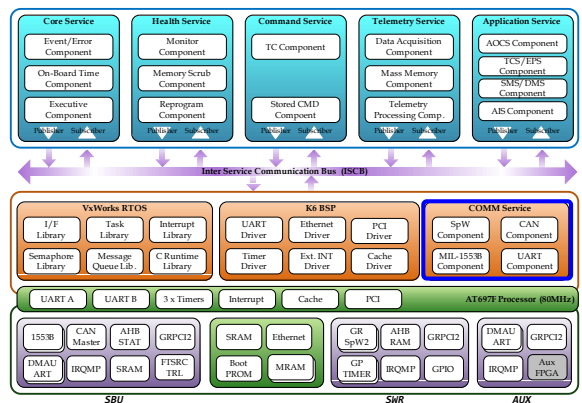


그림 5. Flight Software Architecture

3. 탑재소프트웨어 태스크 아키텍처 및 스케줄링

차세대 저궤도 위성의 탑재소프트웨어는 그림 6과 같이 6개의 태스크로 동작하며 2개의 메인 태스크인 tONE/tTWO 태스크는 가장 높은 우선순위를 가지고 있으며 62.5msec의

sub-minor cycle에 동기화 되어 수행되며 탑재소프트웨어의 실행주기는 125msec인 minor cycle단위로 동작하게 된다. 각 해당 태스크에는 저장명령을 처리하는 Stored Command 컴포넌트와 1553B를 담당하는 1553B 컴포넌트를 제외한 주요 서비스와 컴포넌트가 맵핑된다. 다음 우선순위를 가진 두 개의 1553B 태스크는 even sub-minor cycle 기준으로 1553B 통신이 수행 완료되면 인터럽트에 의해서 깨어나게 되며 해당 minor cycle에서 수행된 1553B 통신의 상태를 확인하고 다음 even sub-minor cycle에 수행될 1553B 통신을 준비한다. tSCP 태스크는 우선순위 130으로 저장명령을 수행 및 처리해야 할 경우 tONE 태스크의 메시지에 의해서 활성화 되는 on-demand 태스크이며, 가장 우선순위가 낮은 tBACK 태스크는 다른 태스크가 동작하지 않을 때 VxWorks에 의해 활성화 되며 오랜 시간이 소요되는 메모리 스크러빙, 대용량 메모리 관리 및 reprogramming 등의 작업을 수행한다.

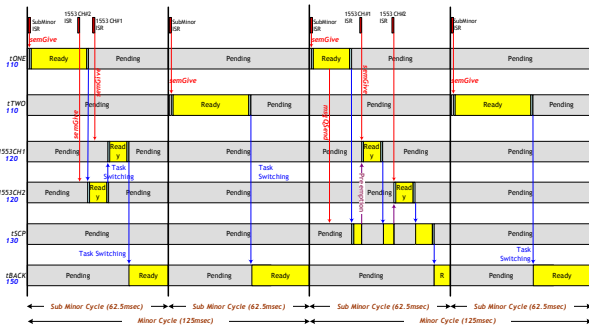


그림 6. FSW Overall Task Scheduling

IV. 탑재소프트웨어 1553B 컴포넌트

1. 탑재소프트웨어 1553B 컴포넌트

차세대 저궤도 위성의 탑재소프트웨어 1553B 컴포넌트는 외부 Gyro, Star Tacker, GPS 수신기 및 탑재체간의 외부 인터페이스를 지원하며, SBU 통신칩에 내장된 2채널의 1553B 통신 모듈을 사용한다. 시스템 소프트웨어 및 1553B 컴포넌트에서는 SBU 통신칩의 SRAM을 통신 데이터 및 Micro Code를 저장하는 용도로 사용하며 AHB I/O 영역에 할당된 1553BRM 컨트롤러를 통해 1553B를 운용한다.

탑재소프트웨어는 그림 7과 같이 SBU PCI BAR2를 통해 SBU의 Actel 1553BRM 코어[6]의 레지스터를 액세스하며, SBU PCI BAR1를 통해 SBU SRAM을 액세스하여 통신 데이터와 Micro Code를 저장한다.

그리고 B1553BRM wrapper[5]를 통해 1553BRM이 AMBA 버스상에서 메모리를 액세스 할 수 있도록 주소를 반드시 설정해주어야 하며, 이때 AT697F는 PCI Initiator로 동작하게 되며 SBU 통신칩은 PCI Target으로 동작한다.

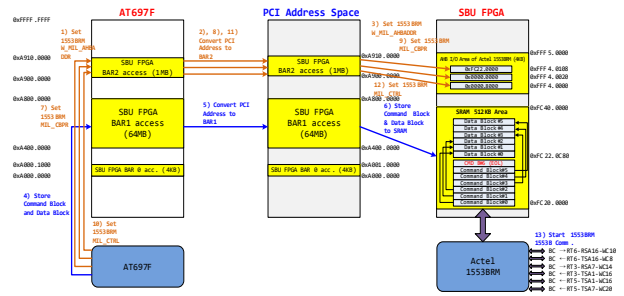


그림 7. FSW 1553B Operation over PCI

모든 설정이 끝나면 탑재소프트웨어에서 1553BRM 코어를 구동하게 되며, 1553BRM 코어는 SBU SRAM에 저장된 Micro Code에 따라 1553B 통신을 수행한다. 정상적으로 1553B 통신이 완료될 경우 EOL (End Of List) 인터럽트가 발생한다. 탑재소프트웨어에서는 1553B EOL 인터럽트가 발생하면 해당 인터럽트 서비스 루틴이 수행되고, 해당 ISR 루틴에서 탑재소프트웨어의 1553B 태스크를 깨우게 된다. 1553B 태스크에서는 해당 even sub-minor cycle에서 수행된 통신 결과를 확인하고, RT → BC 통신의 경우 수신한 데이터를 Global Mapper 영역에 저장하고 다음 even sub-minor cycle에 수행될 1553B 통신을 준비한다. 통신 중 에러가 발생한 경우 1553BRM 레벨에서 에러가 로그되고 1553B 태스크에서는 해당 에러를 로그하며, 통신 에러가 지속적으로 발생하는 경우 상위 Fault Management 레벨로 전달하여 처리한다.

2. Actel 1553BRM IP 코어

그림 8은 Actel 1553BRM이 수행을 보여주는 도식으로 10개의 Actel 1553BRM 레지스터와 3개의 B1553BRM wrapper 레지스터로 1553B 통신을 제어한다. 그리고 Actel 1553BRM은 통신을 위한 Micro Code로 8개의 halfword로 구성된 command block을 사용한다. Command block은 1553BRM이 수행해야 할 명령을 가지고 있는 control word와 1553B 통신을 위한 command word, 1553B 통신에 사용할 data의 offset 주소를 알려주는 data pointer, 그리고 1553B 통신이 수행 결과인 status word를 저장하는 2개의 status word로 구성되며 control word가 CALL/Branch 관련 Opcode인 경우 사용되는 branch address, 그리고 Skip/MFT 관련 Opcode인 경우 사용되는 timer value로 구성된다.

그림 8의 첫 번째 command block의 control word 0x46FE는 1553B BUS-A를 이용한 1553B 통신 수행을 의미하며, command word 0x0820는 BC → RT를 위한 통신으로 RT1-SA01-WC32로 1553B 통신 수행을 의미한다. 그리고 data pointer 0x0640은 0xFC200C80에 저장된 전송 데이터의 data offset 주소를 의미한다. 정상적으로 1553B 통신이 수행될 경우 status word #1에 status word 0x0800이 기록된다. 두 번째 command block의 control word 0x44FE는 1553B

BUS-B를 이용한 1553B 통신 수행을 의미하며, command word 0x0C58은 RT → BC를 위한 통신으로 RT1-SA02-WC24로 1553B 통신 수행을 의미한다. 그리고 data pointer 0x0660은 0xFC200CC0에 수신 될 데이터를 저장할 data offset 주소를 의미한다. 정상적으로 1553B 통신이 수행 될 경우 status word #1에 status word 0x0800이 기록되며 data pointer 주소에 수신 데이터가 기록된다. 마지막 command block은 1553BRM을 정상적으로 종료하는 EOL opcode가 기록되어 있으며 해당 opcode를 수행하면 1553BRM은 정상적으로 operation을 종료하고 EOL 인터럽트를 발생시킨다.

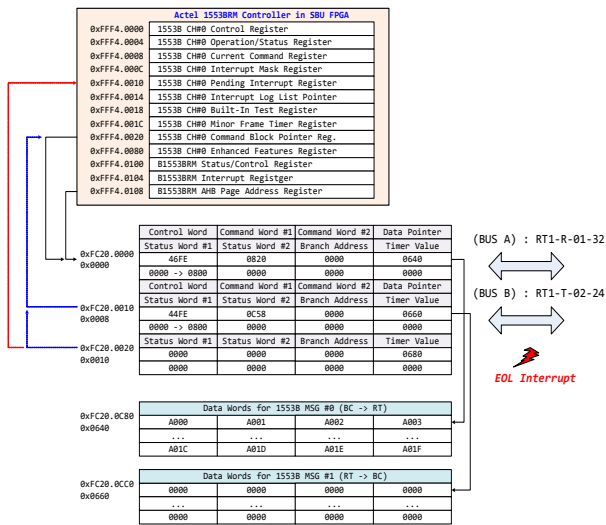


그림 8. Actel 1553BRM Operation

그림 9는 앞에서 설명한 Actel 1553B Operation을 탑재 소프트웨어 시뮬레이터 (FSS, Flight Software Simulator)[7]에서 수행 한 결과를 보여준다. 1553B Monitor를 통해 실제 수행된 1553B 통신 결과를 확인 할 수 있으며, 정상적으로 1553B EOL 인터럽트가 발생한 것을 확인 할 수 있다.

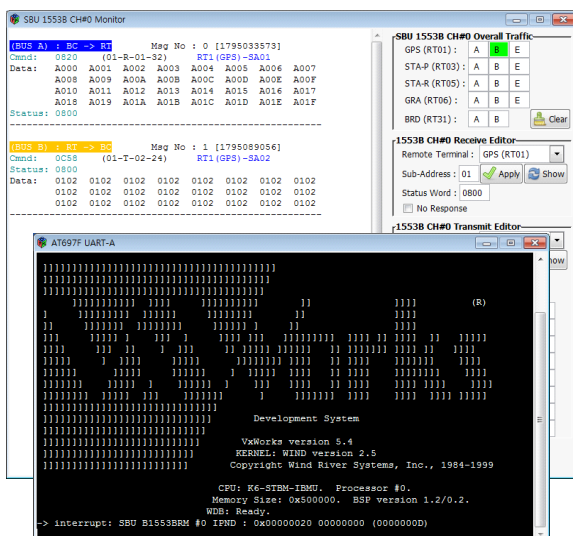


그림 9. Actel 1553BRM Operation Result

3. 탑재소프트웨어 인터럽트 처리방식

차세대 저궤도 위성의 탑재컴퓨터는 3개의 통신칩의 PCI 인터럽트를 처리를 위해 AT697F의 3개의 인터럽트가 할당되어 있으며 1:N 형태로 인터럽트를 처리해야 하기 때문에 vectored interrupt 방식을 사용한다. 그림 10은 현재 차세대 저궤도 위성에서 사용하는 인터럽트 리스트와 연결도를 보여주며, 이중 회색으로 표시된 인터럽트는 polling 방식으로 처리하거나 사용하지 않는 인터럽트를 의미하며, 노란색으로 표시된 인터럽트는 인터럽트가 unmask되어 사용하는 인터럽트를 의미한다. 그리고 하늘색으로 표시된 인터럽트는 지상에서만 활성화 되는 인터럽트를 의미한다.

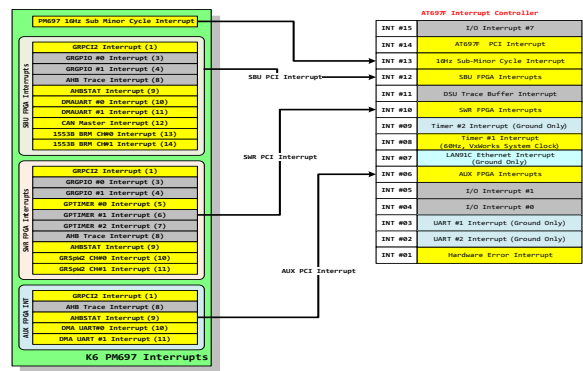


그림 10. FSW Interrupt List and Connection

4. FSW 1553B 인터럽트 처리 방식

각 통신칩의 인터럽트 처리방식은 기존 프로세서 보드에서 처리하는 방식과 다르다. 프로세서 보드에서 발생한 인터럽트 경우 인터럽트 컨트롤러의 IPR (Interrupt Pending Register)에 해당 인터럽트가 기록되고 프로세서가 해당 인터럽트를 인지하게 되면, 자동적으로 IPR의 해당 인터럽트가 클리어 되며 동시에 소프트웨어에 해당 인터럽트가 발생했음을 알리게 된다. 통신칩의 경우 프로세서가 없기 때문에 그림 11처럼 탑재소프트웨어가 IACK (Interrupt Acknowledge) 절차를 반드시 수행해주어야 한다.

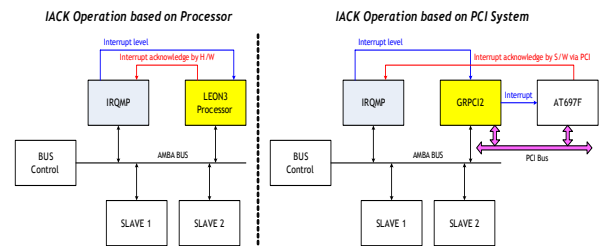


그림 11. IACK Operation Difference

만약 탑재소프트웨어에서 IACK 절차를 수행하지 않을 경우 더 이상 통신칩에서 발생한 인터럽트를 처리 할 수 없게 된다. IACK 절차를 위해 각 통신칩의 IRQMP에 ISA

(Interrupt Status & ACK Register)와 IAD (Interrupt ACK Duration Control Register) 레지스터가 추가적으로 구현되었으며 이를 기반으로 동시에 다수의 인터럽트가 발생하더라도 우선순위 기반으로 인터럽트를 처리할 수 있게 되며 탑재소프트웨어에서는 VxWorks Vectored Interrupt 방식으로 개별 인터럽트를 처리할 수 있게 된다.

각 단계별로 인터럽트가 처리되는 방식을 확인하면 아래와 같은 절차를 통해 처리된다.

- 1) 먼저 SBU Actel 1553BRM 레벨에서 EOL 인터럽트가 발생하게 되면 Actel 1553BRM의 IPR에 EOL 인터럽트가 로그되고, 1553BRM IMASK의 EOL 인터럽트가 unmask가 된 경우 상위 SBU IRQMP에 해당 인터럽트가 전파된다.
- 2) SBU IRQMP 레벨에서 1553B 인터럽트가 unmask된 경우 IRQMP ISA에 1553B 인터럽트 번호를 의미하는 13이 기록되고 상위 SBU GRPCI2로 인터럽트가 전파된다.
- 3) GRPCI2의 DIRQ가 최종적으로 unmask된 경우 AT697F I/O 인터럽트 #5 (인터럽트 #12)가 발생한다.
- 4) AT697F의 인터럽트가 발생하면, AT697F ITP에 해당 인터럽트가 로그된다. AT697F의 ITMP에 인터럽트 #12가 unmask된 경우 AT697F IU에 인터럽트가 전달된다.
- 5) IU의 PSR.PIL (Processor Interrupt Level)이 인터럽트 #12보다 작을 경우 프로세서가 해당 인터럽트를 처리하게 된다. 이와 동시에 AT697F은 IACK 시그널을 생성하여 AT697F ITP에 기록된 인터럽트 #12를 클리어한다.
- 6) AT697F IU가 인터럽트를 수용할 경우 trap이 발생하게 되고, VxWorks를 통해 등록된 탑재소프트웨어 인터럽트 서비스 루틴이 수행되며, SBU IRQMP의 ISA (13)의 값을 읽어와 SBU 1553B CH#1 인터럽트를 처리하는 Sub-ISR 루틴을 수행하게 된다.
- 7) 1553B Sub-ISR 루틴에서는 Actel 1553BRM의 IPR을 읽어와 EOL 인터럽트가 발생한 것을 최종적으로 인지하게 되며, Actel 1553BRM IPR을 초기화 하고, SBU IRQMP ISA에 13을 기록하여 IACK를 수행한다.
- 8) 최종적으로 1553B 태스크에 세마포를 전달하여 1553B 태스크를 깨운다.

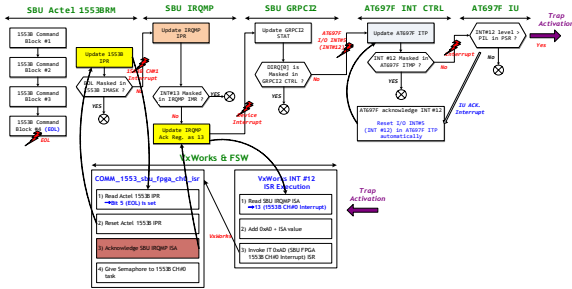


그림 12. FSW 1553B Interrupt Handling

5. FSW 1553B 컴포넌트 테스트

차세대 저궤도 위성의 탑재컴퓨터상에서 탑재소프트웨어를 개발 및 검증하기 위해 그림 13과 같은 STB (Software Test Bed) 환경이 구축되었다. STB에서는 실제 탑재컴퓨터가 사용되며, 다양한 전장품을 위한 시뮬레이터가 사용된다. 또한 지상국을 모사할 수 있는 CORTEX와 지상국 시스템이 동일하게 사용되며, 모든 통신을 모니터링 및 분석 할 수 있는 다양한 툴이 사용된다.

1553B 테스트를 위해 RT 시뮬레이션과 모니터링 기능을 제공하는 PASS3200을 활용하여 1553B 통신이 수행되었다. 기존 탑재소프트웨어 시뮬레이터를 통해서 검증된 소프트웨어를 동일한 방식으로 STB 상에서 구동한 결과 그림 14와 같이 정상적으로 1553B 통신이 수행되었으며 정상적으로 인터럽트가 발생하여 탑재소프트웨어에 의해 처리가 되었다. 1553B 시그널 레벨에서의 검증 결과도 정상적으로 처리되었다.

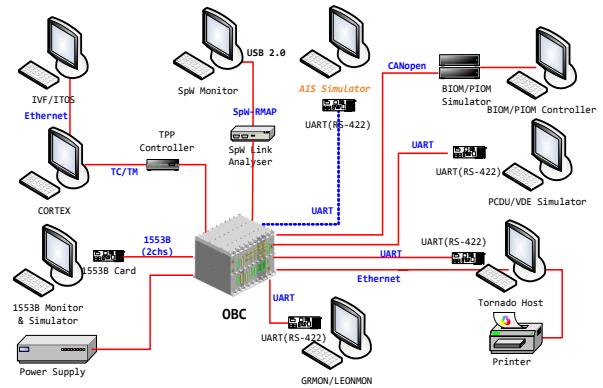


그림 13. Software Test Bed Configuration

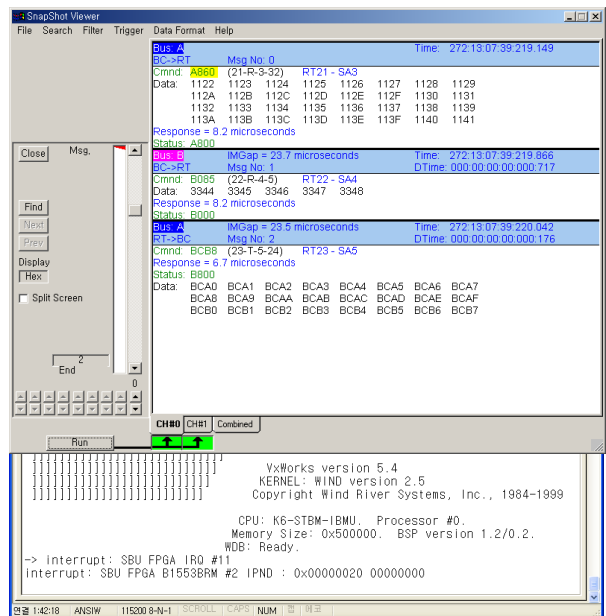


그림 14. 1553B with PCI Test Result

V. 결론

본 논문에서는 현재 항우연에서 개발 중인 차세대 저궤도 탑재컴퓨터의 AT697F 프로세서와 PCI 기반의 탑재컴퓨터 프로세서 보드에 대해 기술하였으며, 탑재소프트웨어 측면에서의 PCI 운영방식과 탑재소프트웨어의 아키텍처에 대해서 설명하였다. 그리고 PCI 기반의 1553B 통신 소프트웨어 컴포넌트의 설계 및 구조에 대해 기술하였으며 인터럽트 처리 방식에 대해서 설명하였다. 이를 기반으로 실제 하드웨어 기반의 STB에서의 테스트 결과 및 탑재소프트웨어 시뮬레이터 기반의 테스트 결과를 제시하였다. 현재 차세대 저궤도 위성의 탑재소프트웨어는 예비설계를 완료하였으며 이를 바탕으로 상세 설계 및 테스트가 진행 중이다.

참고 문헌

- [1] Jong-Wook Choi, Yee-Jin Cheon, "Study of Next Space Processor for Development of Flight Software", KSAS, pp. 809-814, 2012.
- [2] Atmel, "Rad-Hard 32bit SPARC v8 Processor AT697F", Atmel, 2011.
- [3] Yunki Lee, Jihoon Kim, "A Conceptual Study on Standard Architecture Design for the Next Generation Satellite OBC", KSAS, pp. 1018-1024, 2013.
- [4] Jong-Wook Choi, Jae-Yeop Jeong, "Development of System Software based on PCI for Next Generation LEO Satellite", KSAS, pp. 652-657, 2015.
- [5] Aeroflex Gaisler, "GRLIB IP Core User's Manual Version 1.1.0-B4108," 2011.
- [6] Actel, "Core1553BRM v3.1 Handbook," 2010.
- [7] Jong-Wook Choi, Yee-Jin Cheon, "Development Status of Processor Emulator and Spacecraft Simulator for Flight Software Development", KSAS, pp. 775-782, 2015.
- [8] Jiri Gaisler, "LEON-1 Processor - First Evaluation Results," European Space Components Conference (ESCCON), Vol. 439, pp.183~187, 2000.
- [9] The ESA Next Generation Microprocessor (NGMP), <http://microelectronics.esa.int/ngmp/>

저자

최 종 욱(Jong-Wook Choi)



- 1999년 2월 : 경북대학교 전자공학 학사졸업
- 2001년 2월 : 경북대학교 전자공학 석사졸업
- 2016년 2월 : 충남대학교 컴퓨터공학과 박사 졸업

· 2000년 12월 ~ 현재 : 한국항공우주연구원 위성비행소프트웨어팀 선임연구원

<관심분야> : 시뮬레이터, 실시간운영체제

정희원

정 재 엽(Jae-Yeop Jeong)



- 2007년 2월 : 충남대학교 컴퓨터공학 학사졸업
- 2009년 2월 : 충남대학교 컴퓨터공학 석사졸업
- 2009년 1월 ~ 2013년 12월 : LIG넥스원 항공연구센터 선임연구원

· 2000년 12월 ~ 현재 : 한국항공우주연구원 위성비행소프트웨어팀 선임연구원

<관심분야> : 임베디드시스템, 실시간운영체제

정희원

유 범 수(Bum-Su Yoo)



- 2009년 2월 : 한국과학기술원 전기 및 전자공학 학사졸업
- 2011년 2월 : 한국과학기술원 전기 및 전자공학 석사졸업
- 2016년 2월 : 한국과학기술원 전기 및 전자공학 박사졸업

· 2015년 12월 ~ 현재 : 한국항공우주연구원 위성비행소프트웨어팀 선임연구원

<관심분야> : 임베디드시스템, 로봇틱스