

SMCS Chip Set 및 소프트웨어 제어절차 분석

채동석^o 이재승 최중욱 이종인, 김학정
 한국항공우주연구원 위성전자그룹
 {dschae^o, jslee, jwchoi, jilee, hjkim}@kiss.or.kr

A Study of SMCS Chip Set and S/W Control Procedure

Dongseok Chae^o Jaeseung Lee Jongwook Choi Jongin Lee Hakjung Kim
 Satellite Electronics Dept., Korea Aerospace Research Institute

요 약

인공위성 탑재컴퓨터의 내부 인터페이스를 위하여 SpaceWire 표준을 적용한 SMCS Chip Set의 사용이 고려되고 있다. SpaceWire는 IEEE-1355 프로토콜을 적용한 것으로 위성체 내에서 다양한 모듈들 간에 손쉬운 표준 인터페이스를 제공한다. 또한 다수의 모듈간의 상호 교차 연결을 위한 Cross-strap 인터페이스 구현이 간단하게 구현될 수 있으므로 위성 운용 기간 중의 높은 신뢰도를 보장할 수 있다. 본 논문에서는 SpaceWire 표준을 적용한 SMCS Chip Set에 대한 소개와 SMCS Chip Set 통한 데이터 전송에 필요한 소프트웨어 제어절차에 대해서 기술하였다.

1. 서 론

인공위성 탑재컴퓨터의 내부 인터페이스를 위하여 유럽(ESA) 표준으로 자리 잡고 있는 SpaceWire 표준을 사용할 계획이다. SpaceWire는 IEEE-1355 프로토콜을 적용한 것으로 모든 Node가 Master가 될 수 있으며, 400 Mbps까지의 고속의 데이터 전송이 가능하므로 위성체 내에서 다양한 모듈들 간에 손쉬운 표준 인터페이스를 제공한다. 또한 node를 손쉽게 추가 혹은 제거할 수 있도록 Cold-Standby 기능을 제공하고 다수의 모듈간의 상호 교차 연결을 위한 Cross-strap 인터페이스 구현이 간단하게 구현될 수 있으므로 위성 운용 기간 중의 높은 신뢰도를 보장할 수 있다[1].

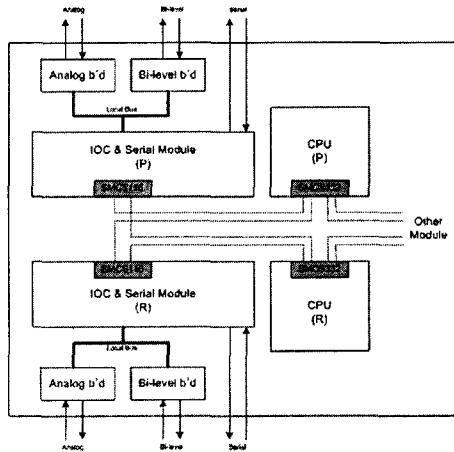


그림 1. 위성 탑재컴퓨터 내부통신 연결

SpaceWire 표준을 적용한 대표적인 우주용 부품인 SMCS(Scalable Multichannel Communication Subsystem) Chip Set이다. 그림 1은 차세대 위성컴퓨터에서 사용 예정인 내부 모듈 간 통신연결의 일부를 나타낸 것이다. CPU 모듈과 IOC(I/O Controller) & Serial 모듈 사이에 SMCS332와 SMCS116를 연결하여 Analog, Bi-level 등의 I/O 데이터를 제어하는데 사용할 계획이다 [2]. 그리고 다른 모듈과의 인터페이스에도 확장하여 유사하게 적용될 수 있다. 본 논문에서는 SMCS Chip Set인 SMCS332, SMCS116의 구성과 주요 특징에 대해서 기술하였고, SMCS Chip Set 통한 데이터 전송에 필요한 소프트웨어 제어절차에 대해서 기술하였다.

2. SMCS Chip Set

2.1 SMCS332

SMCS332는 3개의 Link 채널, COMI (Communication Memory Interface), HOCI (Host Control Interface), PRCI(Protocol Command Interface) 등으로 구성되어 있다. 3개의 Link Channel은 DS-Link macro cell, FIFO를 포함한 Receive/Transmit, Protocol Processing Unit으로 구성되고, 1.25Mbps에서 최대 200Mbps까지 전송 속도를 선택할 수 있다. COMI는 통신메모리를 통한 데이터 전송을 위한 것으로 SMCS332가 CPU와 DPRAM (Dual Port RAM)을 공유하는데, CPU는 단지 통신메모리에 데이터를 기록하고 관련 레지스터를 설정함으로써 데이터를 전송할 수 있는 매우 효율적인 방법이다. HOCI는 주로 SMCS 내부 레지스터를 Access 하기 위하여 설계되어졌는데, HOCI를 통하여 CPU가 직접 SMCS 내부 FIFO를 이용하여 작은 패킷의 데이터를 전송할 수도 있다[3].

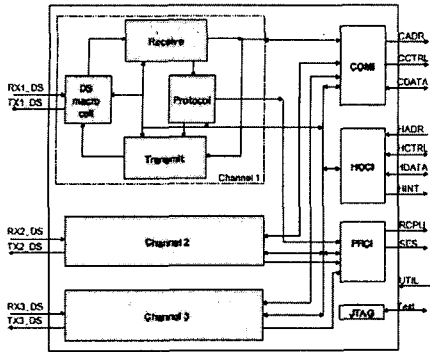


그림 2 SMCS332 구성도

운용모드에는 프로토콜 포맷에 따라서 Transparent 모드, Protocol 모드가 있다. Transparent 모드는 두 노드 사이에 SMCS의 아무런 중재 없이 데이터를 전송하는 것으로 EOP(End Of Packet)를 통해 데이터 수신이 완료되었음을 알 수 있다. SIC(Simple Inter-processor Communication) Protocol 모드는 프로토콜 규격에 따른 SIC Protocol을 사용하는 것으로 Protocol의 헤더로 4개의 데이터 토큰을 사용한다.

2.2 SMCS116 (SMCSlite)

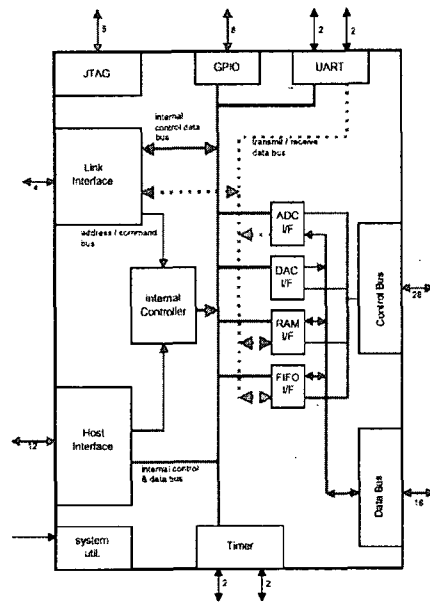


그림 3 SMCS116 구성도

SMCS116은 기존의 SMCS332에서 3개의 Link를 사용하는 대신에 1개의 Link만을 사용하여 주로 원격으로 제어되도록 설계되었다. ADC(Analog to Digital Converter), DAC(Digital to Analog Converter), RAM,

FIFO(First In First Out) 등의 I/F를 제공한다. 그림 4는 SMCS116을 응용한 예를 나타낸 것이다.

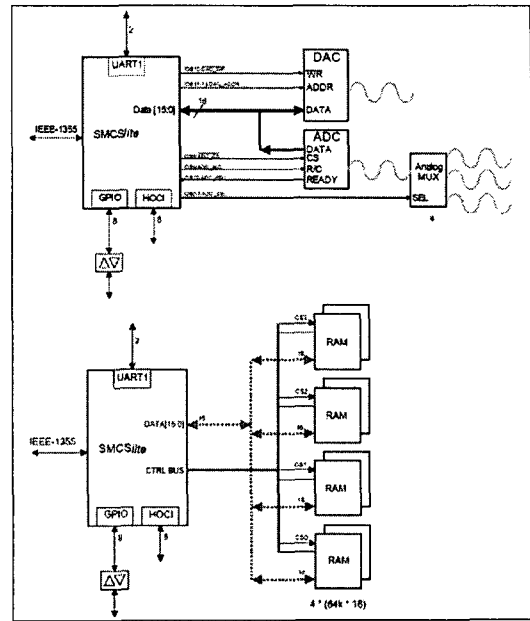


그림 4 SMCS116 응용 예

SMCS116 내부 레지스터의 프로그래밍은 1355 link를 통한 프로토콜에 의해서 또는 Host I/F를 통하여 수행할 수 있다. 레지스터는 1 byte를 포함하는 반면 포트는 FIFO와 같이 Multiple 데이터를 포트를 통하여 read/write 할 수 있다. 그림 5는 포트 통하여 데이터 전송 포맷을 나타낸 것이다[4].

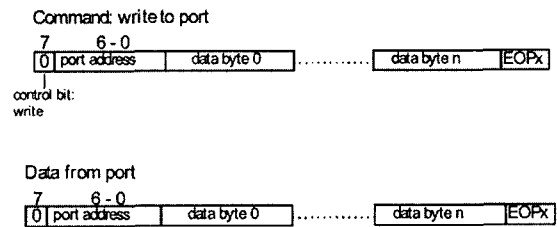


그림 5 포트를 통한 데이터 전송

3. 소프트웨어 제어절차

다음은 그림 1과 같이 SMCS332와 SMCS116가 연결되어 SMCS332는 CPU에 의하여 제어가 되고 SMCS116은 SMCS332와 Link로 연결되어 Control by Link 방식으로 제어되는 경우에 필요한 제어 절차를 나타낸 것이다. SMCS116과 I/O Module과의 연결은 RAM I/F 방식을 적용하여 RAM Address에 각 I/O가 연결된다. 초기화 과정과 COMI를 통한 Data 전송 절차는 다음과 같다.

3.1 초기화 과정

전체적인 제어 순서는 먼저 SMCS332의 레지스터를 설정하고, SMCS116과의 Link를 연결한다. Link가 설정되면 Link를 통하여 SMCS116의 레지스터를 설정한다. Link 연결은 SMCS332에서 Link를 통하여 NULL을 송신함으로써 시작된다.

3.2 데이터 전송절차

SMCS332 COMI를 통하여 SMCS116에 데이터를 송신하는 절차는 다음과 같다.

- ① 전송할 데이터 포맷하여 통신메모리에 저장한다.
- ② SMCS116에서 데이터를 수신하도록 SMCS116 레지스터 설정한다. 해당 레지스터에 데이터가 기록되어야 할 시작 번지와 마지막 번지를 기록한다.
- ③ 통신메모리에 저장된 데이터를 송신하도록 SMCS332의 레지스터를 설정한다. 해당 데이터가 기록된 시작 번지와 마지막 번지에 해당하는 레지스터를 설정한다.
- ④ SMCS332 인터럽트 상태 레지스터를 통하여 전송결과를 확인한다.

SMCS116으로부터 데이터 수신절차는 다음과 같다.

- ① 전송해야 할 데이터 영역의 시작 번지와 마지막 번지를 지정하기 위해 SMCS116 레지스터를 설정한다.
- ② 통신메모리에 수신할 데이터를 저장하기 위하여 SMCS332의 레지스터를 설정한다. 해당 데이터가 저장될 시작 번지와 마지막 번지를 설정한다.
- ③ 데이터 송신을 시작하도록 SMCS116 레지스터를 설정한다.
- ④ 인터럽트 상태레지스터를 통해 수신결과를 확인한다.
- ⑤ 통신메모리에 저장된 데이터를 읽는다.

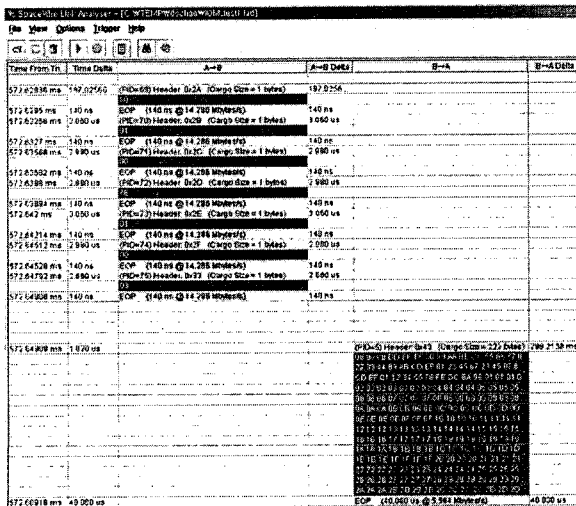


그림 6 COMI를 통한 데이터 수신

위의 제어절차에 따라 탑재컴퓨터 DM(Development Model)을 이용하여 데이터 전송시험을 수행하였다. CPU에서 통신메모리에 전송할 데이터를 기록한 후 SMCS332 COMI를 통하여 SMCS116에 데이터를 송신하여 해당 I/O 영역에 데이터를 기록하고, 동일한 I/O 영역을 SMCS116에서 SMCS332로 송신하도록 시험을 수행하였는데, 그림 6은 모니터링 도구인 SpaceWire Link Analyser를 통하여 SMCS116으로부터 데이터를 송신하도록 제어하는 과정을 포함하여 SMCS332로 데이터를 송신하는 부분을 도시한 것이다. 전송받은 데이터는 마지막 부분인데, 앞부분 7번의 데이터 전송은 Link를 통하여 SMCS116에서 데이터를 전송하도록 레지스터를 설정하기 위해 필요한 부분이다.

표 1은 100Mbps 통신 속도를 사용하였을 경우 I/O 데이터를 전송하는데 필요한 모든 제어 절차를 포함하여 전체 소요되는 시간을 측정하는 것이다.

표 1 데이터 전송시간

Data Size (Byte)	Transmit Time (usec)	
	Write	Read
2	32.7	28.5
50	70.5	82.3
100	106.7	134.7
160	154.5	189.7
220	195.2	251.7

4. 결론

SpaceWire 표준을 적용한 대표적인 우주용 부품인 SMCS Chip Set(SMCS332, SMCS116)의 구성과 주요 특징에 대해서 기술하였고, 인공위성 탑재컴퓨터 내부 모듈 간 SMCS Chip Set을 연결하여 사용할 경우, 데이터 전송에 필요한 소프트웨어 제어절차에 대해서 기술하였다. 그리고 이러한 절차를 통하여 통신을 수행한 결과 데이터 전송이 이상 없이 수행됨을 확인할 수 있었다. 향후 HOCI를 통한 데이터 전송 및 SIC Protocol Mode를 이용한 데이터 전송 등에 대한 분석이 수행하여 여러 가지 방식 중에 가장 최적의 통신방식을 적용할 계획이다.

참고문헌

- [1] 김대영, "SpaceWire 표준과 차세대 위성 개발에 대한 적용", 한국항공우주학회 추계학술대회, 2004
- [2] 권기호, "IO Board Design of Next Generation Satellite using the SpaceWire Interface", 한국우주과학회, 2004
- [3] DSS, "SMCS332 User Manual", 1999.
- [4] A.Z.Christen, "SMCSlite User Manual", 2001.
- [5] <http://www.estec.esa.nl/tech/spacewire/>