

## 우주 탐사선에서 파일 시스템의 역할 및 응용 기술 연구

구철희\*, 주광혁\*\*

## A study on the role and application technology of the space explorer

Cheol-Hea Koo\*, Gwang-Hyeok Ju\*\*

## Abstract

Computing environment of space explorer including LEO (Low Earth Orbit), GEO (Geosynchronous Earth Orbit) satellite may be considered as the same category of embedded system on the ground. But with comparison with personal computing environment it is widely accepted that the space computing is outdated and behind of state of the art. Especially file system which is nearly essential item in all ground computing environment including personal computer, workstation and server is rarely used in space explorer till lately. In this paper, a study of ESA PUS (Packet Utilization Standard) and CCSDS (Consultative Committee for Space Data Systems) community's activity, international trend, and applicable technical application for applying file system in order to use these standards for utilizing the file system to deep space explorer is described.

## 초 록

LEO (Low Earth Orbit), GEO (Geosynchronous Earth Orbit) 위성을 비롯한 우주 탐사선의 컴퓨팅 환경은 지상의 내장형 시스템(Embedded System)과 동일한 범주로 간주될 수 있으며 개인용, 산업/서버용 컴퓨터와 비교했을 때 비교적 원시적이고 발전 속도도 더뎠다. 특히 개인 컴퓨터 및 서버/워크스테이션에서 필수적으로 사용되고 있는 파일 시스템도 우주 탐사선에는 최근까지 거의 사용되지 않았다. 본 논문에서는 최근에 우주 탐사선에 파일 시스템을 적용하기 위한 유럽 PUS (Packet Utilization Standard)와 CCSDS (Consultative Committee for Space Data Systems) 커뮤니티의 관련 규격 제정 등 국제적 연구 개발 동향 및 우주 탐사선에 적용가능한 응용 기술 분야에 대한 연구 결과를 기술한다.

키워드 : 파일 시스템 (file management system), 운용 소프트웨어 (operation software), 임무 운용 (mission operation), 비행 소프트웨어 (flight software), CFDP (CCSDS File Delivery Protocol)

## 1. 서 론

인류가 컴퓨터를 접한 이후 저장장치는 컴퓨터의 필수적인 구성요소가 되었다. 특히 휘발성인 RAM (Random Access Memory) 저장장치와 다르게 비휘발성 저장장치의 발전은 매우 눈부셨다. 플로피 디스크으로 시작된 저장장치는 곧 광디스크, 하드디스크 및 플래쉬 디스크로 발전해 갔으며 그 용량 및 처리 속도는 해마다 커지고 빨라지고 있다. 그 이면에 내장형 시스템 (Embedded System)의 경우처럼 제한된 리소스에서 정확히 정의된 작업을 수행하기 위해서 저장장치의 사용이 최소화된 경우도 있었다. 위성을 비롯한 우주 탐사선의 경우가 좋은 예이다.

1980년대의 우주 탐사선은 컴퓨터가 없이 디지털 및 아날로그 처리 회로만 가지고 운용되었고 1990년대 들어서 본격적인 컴퓨터를 가진 우주 탐사선의 시대가 열리게 되었다. 하지만 이 초창기 우주 컴퓨터들은 제한된 RAM 및 ROM (Read Only Memory), 플래쉬 (Flash) 저장장치만을 가지고 전형적인 내장형 시스템의 기능을 수행했지만 요구사항의 복잡화, 운용의 편리함 추구 등의 이유로 이들 고전적 내장형 시스템은 본격적인 변화의 요구를 맞이하게 되었다.

## 2. 우주용 컴퓨터 시스템의 역사

### 2.1 개인/서버/워크스테이션 컴퓨터

개인 컴퓨터에서 파일 시스템은 유저가 사용하거나 생성한 데이터를 보관하는데 필수적으로 사용되며 각각의 운영 체제(Operating System, OS)에 따라서 (표 1)과 같이 다양한 파일 시스템이 개발되어 사용되고 있다<sup>1)</sup>.

초기의 저장 장치는 용량이 매우 작았기 때문에 파일 시스템도 단순했지만 현재는 TB (tera byte, 10<sup>12</sup>)크기의 저장장치가 일반화 되어 있으며 파일 시스템도 오류 복구를 위한 저널링을 지원하는 등 복잡해 졌다. 파일 시스템은 운용 체제를 로딩하고 유저 상태를 저장하고 원격의 저장소로부터 정보를 가져오거나 저장하는데 필수

적으로 사용된다. 예를 들어 MS 윈도우 운영 체제에서 레지스트리 (Registry) 라는 파일을 통해서 모든 운영 체제의 관리를 하고 있는 것처럼 파일이 없다면 현재와 같은 편리한 컴퓨팅 환경은 상상도 할 수 없을 것이다.

표 1. 파일 시스템 현황

운영체제	파일 시스템
MS Dos/Windows	FAT12/16/32, NTFS
Unix/Linux	ext2/3/4, XFS, JFS, YAFFS
OSX	Mac OS, HFS Plus

### 2.2 우주용 컴퓨터

한국의 최초 통신방송위성이고 미국 Lockheed Martin에서 개발한 무궁화위성 1, 2호 (Koreasat-1, 2)는 컴퓨터가 탑재되지 않고 모두 FPGA (Field Programmable Gate Array) 기반의 디지털 처리 모듈에서 원격명령 처리 및 텔레메트리 생성 및 송신을 담당했다. 이후 발사된 무궁화위성 3호(Koreasat-3, Lockheed Martin 사 개발, A2100 버스)에는 미국 하니웰 사 (Honeywell, FL USA)가 개발한 HSC (Honeywell Space Computer)를 탑재했으며 프로세서는 1750 프로세서였다.

표 2와 같이 우리별 1,2를 시작으로 한국 최초의 실용 관측위성인 아리랑 다목적 위성 시리즈 까지 모두 컴퓨터를 탑재하고 실시간 운영 체제 (Real-Time Operating System, RTOS)를 사용해 위성을 운영하였다.

2009년에 발사된 LRO (Lunar Reconnaissance Orbiter)는 BAE의 RAD750 컴퓨터를 탑재하고 있었고, 총 384 Gigabits (48 GBytes)의 용량을 제공하는 4개의 cPCI 타입 DSB (Data Storage Boards)와 FAT32 파일 시스템을 사용하였다. 이외에 NASA는 LRO의 시스템 부트 (System Boot)와 미션 구성 데이터 (Mission configuration data)를 위해 자체 제작한 단순한 파일 시스템 또는 인-메모리 파일 시스템 (In-memory file system)을 사용하였다. NASA

GSFC는 LRO에 cFE(core Flight Executive)/CFS(Core Flight System) 소프트웨어 플랫폼을 처음 적용하였으며 2011년 오픈 소스로 공개하였다. cFE/CFS 소프트웨어 플랫폼은 파일 시스템을 활용하고 있다.

2003년에 발사된 Mars Express는 총 12 Gbit 제공하는 SSMM (Solid State Mass Memory)를 가지고 있으며 FAT16 파일 시스템을 사용하여 데이터를 관리하였다<sup>(2)</sup>.

표 2. 한국 위성 컴퓨터 탑재 현황

위성 프로젝트	발사 연도	프로세서	RTOS	파일 시스템
우리별 1,2 KITSAT-1, 2	1992	80C186	SCOS	FAT
우리별 3	1999	80960	Nucleus	FAT
KOMPS AT-1	1999	80186	VRTX	-
STSAT-1	2003	80960	Nucleus	-
KOMPS AT-2	2006	80386	VRTX	-
STSAT-2	-	PPC603e	VxWorks	-
DubaiSat-1	2009	ERC32	RTEMS	FAT
COMS	2010	1750A	ASTRES	-
KOMPS AT-3	2012	ERC32	VxWorks	-
STSAT-2C	2013	leon3	RTEMS	-
KOMPS AT-5	2013	ERC32	VxWorks	-
DubaiSat-2	2013	LEON3	VxWorks	FAT

### 3. 국제 표준

위성 관련 통신 규격에서 파일 시스템에 대해서 규정하고 있는 것은 크게 ESA PUS (Packet Utilization Standard)와 CCSDS (Consultative Committee for Space Data Systems)로 대표되며

파일 시스템의 사용 목적은 유사하다. 이들 통신 규격은 파일 시스템을 강제하는 성격의 규격이 아니라 파일 시스템을 사용했을 때 가장 성능을 극대화할 수 있는 규격이다.

#### 3.1 ESA PUS

그림 1에서 볼 수 있듯이 Service 13 "Large data transfer (LDT) service", Service 15 "Data storage and retrieval service"가 온-보드 파일 처리와 관련된 서비스이다.

Service Type	Service Name
1	Telecommand verification service
2	Device command distribution service
3	Housekeeping & diagnostic data reporting service
4	Parameter statistics reporting service
5	Event reporting service
6	Memory management service
7	Not used
8	Function management service
9	Time management service
10	Not used
11	On-board operations scheduling service
12	On-board monitoring service
13	Large data transfer service
14	Packet forwarding control service
15	On-board storage and retrieval service
16	Not used
17	Test service
18	On-board operations procedure service
19	Event-action service

그림 1. PUS services (ESA ECSS-E-70-41A)

#### 3.2 CCSDS

파일 시스템 관련한 규격은 CFDP (CCSDS File Delivery Protocol)<sup>(4)</sup>, Proximity-1<sup>(5)</sup>, 그리고 SOIS (Spacecraft Onboard Interface Services)<sup>(6)</sup>이다.

한 예로 그림 2에서 볼 수 있듯이 CFDP는 유선과 통신 시스템 사이에 파일 저장소가 있고 통신 흐름은 CFDP 프로토콜에 의해서 제어된다. 흐름 제어는 PDU (Protocol Data Unit)에 의해서 수행되고 파일 데이터는 FDU (File Delivery Unit) 단위로 지상에서 나뉘어져 송출되고 위성은 나뉜 FDU를 모두 받아서 하나의 파일로

완성하며 무결성(integrity) 체크를 수행한다. 손실된 FDU가 있다면 다시 전송(retransmission)해 달라는 PDU를 송신측에 보내 재전송을 요청한다. CFDP는 아래 우주 탐사 프로젝트에서 사용되었다.

- 2001 NASA Odyssey
- 2003 ESA Mars Express
- 2005 NASA MRO (Mars Reconnaissance Orbiter)
- 2009 NASA LRO
- 2009 NASA MER (Mars Exploration Rover)
- 2011 NASA MSL (Mars Science Laboratory, Curiosity)

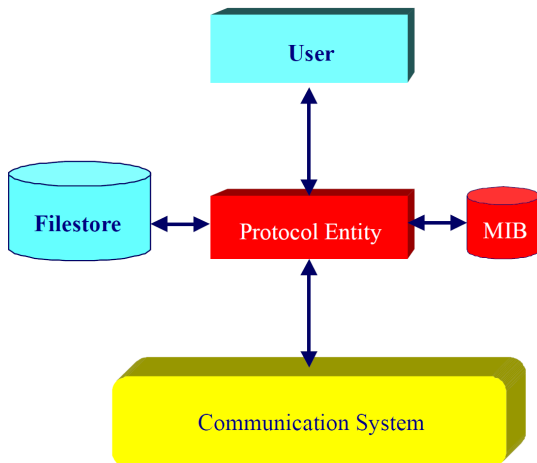


그림 2. CFDP 아키텍처 (CCSDS 727.0-B-4)

SOIS는 CCSDS에서 표준화하고 있는 온-보드 소프트웨어 아키텍처이며 그림 3과 같이 응용 지원 계층 (Application Support Layer)과 서브-네트워크 계층 (Subnetwork Layer)에 여러 서비스로 구성되어 있으며 FPSS (File and Packet Store Service)는 응용 지원 계층에 존재한다.

ESA PUS와 CCSDS 표준에서 정의한 공통적인 파일 서비스의 기능은 아래와 같다<sup>(7)</sup>.

- Data transport  
: 저수준 통신 프로토콜

- Files transfer  
: 파일 전송 프로토콜 및 관련 관리 기능
- Files management  
: 파일/디렉토리 생성, 복사, 이동, 삭제 등의 기본 파일/디렉토리 관리 시스템
- Files utilization  
: 지상국 또는 온-보드에서 위성 운영시 파일의 사용이 가능하도록 인터페이스가 제공되어야 함

위에서 언급된 필요 기능 중 파일 전송 프로토콜은 CFDP와 같은 DTN 프로토콜에서 제공하며 나머지 부분 중 보통 실시간 운영체제가 지원하는 파일 관리 시스템(File Management System, FMS)에 있는 기능이 아니라면 자체 개발 또는 SOIS 아키텍처를 고려 가능하다.

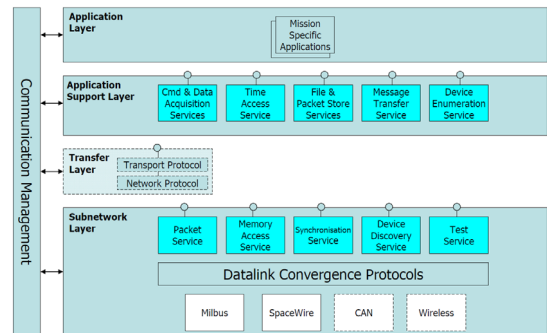


그림 3. SOIS 아키텍처 (CCSDS 873.0-M-1)

### 3.3 실시간 운영 체제 별 FMS

큐브 위성(Cubesat)을 포함한 위성에 사용되고 있는 실시간 운영체제는 매우 다양하지만 본 논문에서는 WindRiver VxWorks와 OAR RTEMS에 포함된 FMS에 대해서 조사해 보았다. 표3에는 VxWorks 및 RTEMS의 경우 실제 위성을 비롯한 우주탐사에 사용되었거나 적용 가능한 RAM 또는 EEPROM의 FMS가 정리되어 있다. 특별히 플래시 기반 저장 장치(Solid State Recorder, SSR)를 사용할 경우 특정 공간에 계속 쓰기 (Write)해서 생기는 칩의 성능 저하를 예방하는 웨어-레벨링 (wear leveling) 기술을 고려하

여야 한다.

표 3. RTOS 별 FMS

	VxWorks	RTEMS
EEPROM	ROMFS <sup>3)</sup>	IMFS <sup>1)</sup> + tar
RAM	MSDOS (FAT16)	RFS <sup>2)</sup>
SSR	MSDOS (FAT16) TrueFFS <sup>4)</sup>	MSDOS RFS

- 1) In Memory File System
- 2) RTEMS File System
- 3) Read-Only Memory File System
- 3) Flash Wear Leveling

## 4. 운용 방법

위성을 비롯한 우주 탐사선에 파일 시스템이 도입된 이후로 온-보드 데이터 처리 및 온-보드 소프트웨어 수행의 많은 변화가 가능해졌다.

### 4.1 Science data playback

LRO의 예에서 볼 수 있듯이 우주 탐사선에서 파일 시스템은 수십 Gbyte에 달하는 SSR에 보관되어 있는 과학 데이터를 효과적으로 관리하기 위해서 요구되었다.

표 4와 같이 LRO는 여러 개의 탑재체를 가지고 있으며 약 50 Gbytes의 SSR을 가지고 있었다. 따라서 이들 데이터의 관리에 파일 시스템의 사용이 반드시 필요했으며 지상으로의 플레이-백(playback)은 CFDP를 통해 수행되었다.

탑재체에 의해서 수집된 과학 데이터의 저장 및 플레이-백은 초창기 파일 시스템의 주목적이었다.

### 4.2 Data relay

심우주 탐사선(착륙선 및 로버 포함)은 지구와의 통신을 위해서 반드시 심우주 통신 네트워크(Deep Space Network, DSN)를 활용해야 한다. 예를 들어 화성과 지구가 최단 거리에 있을 때

전파가 도달하는데 약 14분이 소요되어 통신 신호 세기가 매우 약하기 때문에 중간에 신호를 증폭하는 릴레이 위성이 필요하다. 또한 지구-달의 경우에도 자전과 공전 때문에 항상 LOS(Line Of Sight)를 확보하기 위해서는 지구에 최소 3개의 대형 안테나(NASA JPL의 DSN 안테나는 직경 34 m 급)가 120도 간격으로 설치되어 있어야 한다.

표 4. 미국 NASA LRO (2009) 탑재체

탑재체	설 명
CRaTER	Cosmic Ray Telescope
Diviner	Diviner Lunar Radiometer Exp
LAMP	Lyman-Alpha Mapping Project
LEND	Lunar Explorer Neutron Detector
LOLA	Lunar Orbiter Laser Altimeter
LROC	WAC (Wide-Angle Camera) NAC (Narrow-Angle Camera)
Mini-RF	Synthetic Aperture Rader

심우주 통신의 경우 릴레이 위성이 bent-pipe 방식의 중계가 불가능하기 때문에 BP (Bundle Protocol, RFC-5050) 또는 LTP (Licklider Transmission Protocol, RFC-5326)와 같은 DTN (Delay 또는 Disruption Tolerant Network) 프로토콜의 경우 모두 “Store & Forward” 방식을 사용한다. 이 방식은 중계하고자 하는 데이터를 우선 중계 위성에 저장(보관)한 후 다음 중계 지점과의 통신이 확보되면 데이터를 전달한 후 자신이 갖고 있던 데이터 사본을 삭제하는 방식이다. 따라서 데이터 릴레이의 경우 파일에 기반을 둔 시스템으로 구성되어야 복잡도를 최소화할 수 있다. RAM은 통신 시스템과 저장 장치 사이의 중간 단계의 임시 파일 보관소로 활용된다.

### 4.3 TC/TM 처리

과학 데이터 전송 및 데이터 릴레이를 수행하기 위해서 구축된 온-보드 FMS은 자연스럽게 원격명령(TC) 및 텔레메트리(TM) 처리에 적용되게 되었다.

지구 저궤도(LEO) 및 정지궤도(GEO) 위성의 경우 TC, TM은 대부분 ESA PSS 표준(한국의 경우 천리안 위성) 및 CCSDS 표준(한국의 경우 다목적 위성)에 의해서 처리되었으나 우주 탐사선은 항상 지구와의 LOS가 보장되지 않기 때문에 TC 및 TM도 일반 과학 데이터처럼 처리하는 것이 위성 운용의 신뢰성을 높일 수 있다는 것이 밝혀졌다. 메모리 덤프(memory dump)와 같은 TM 동작은 파일 시스템을 적용했을 때 심우주 통신을 고려하면 이상적인 운용이 가능하다.

특히 소프트웨어 패치(on-board software patch) 또는 미션 파라미터 업데이트(mission parameter upload/update) 등 상대적으로 큰 사이즈의 데이터를 업-로드할 경우 파일 시스템을 사용하여 이어받기 기능을 적용할 수 있다.

VCID의 경우 TM에는 적용가능하나 TC에는 적합하지 않고, MAP ID의 경우 TM에는 적합하지 않다. 하지만 CFDP의 APID는 TC 및 TM 모두에 적용가능하다<sup>(8)</sup>.

#### 4.4 위성 운영

온-보드 파일 시스템의 사용의 주된 목적은 가능한 한 고수준의 자동화(Autonomy)를 우주 탐사 미션에 제공하는 것이다. 비록 온-보드 파일 시스템을 위성 운영에 사용하기 위한 일반 표준은 만들어지지 않았지만 유럽, 미국 등의 우주 탐사에서 효율적이고 검증된 방법이 고안되고 적용되어 오고 있다<sup>(7,8,9)</sup>.

예를 들어 위성의 모드, 시간, 매크로, 또는 구성 정보 등을 패킷 단위로 전송하는 것보다는 파일로 하나로 묶어 전송하고 파일로 읽어들이는 것이 덜 복잡하다. 왜냐하면 파일로 전송하는 방식은 치밀한 ICD(Interface Control Drawing) 설계 또는 문서의 유지보수를 필요로 하지 않기 때문에 개발 시간을 덜 소요하면서도 보다 독립적으로 해당 기능을 운영 가능하다.

과학 데이터를 측정하고 처리까지 해야 하는 탑재체를 가진 우주 탐사선(특히 로버)이라면 파일 시스템은 전체의 작업을 몇 개의 독립적인 단위로 분리가능하게 한다. 예를 들어 센서는 암석

등 시료에서 데이터를 추출하여 파일로 생성하여 온-보드 컴퓨터에 보내고 온-보드 컴퓨터에 탑재된 온-보드 소프트웨어는 파일이 완료되었을 때 파일 생성 시간 순서대로 처리하는 것이 가능해진다. 이 과정에서 복잡한 패킷 처리의 부담을 덜 받는 장점이 있다<sup>(10)</sup>.

또한 지상국에서 위성에 업로드 할 파일의 내용을 간단히 살펴볼 수 있기 때문에 지상국의 인력이 저수준 작업보다 좀 더 분석적이고 운용적인 측면에 집중할 수 있게 된다.

##### 4.4.1 온-보드 소프트웨어 관리

LRO의 경우 NASA는 PROM에 파일 시스템을 사용하여 부트 이미지(boot image)를 초기 부트-업 시간에 로드할 때 파일로부터 읽어 드렸다<sup>(11)</sup>. 이는 TC 업링크 또는 SGM(Safe Guide Memory)을 통해 파일로 소프트웨어 이미지가 주어지면 일관되고 쉬운 방법으로 새로운 소프트웨어 이미지로 컴퓨터를 다시 기동할 수 있음을 의미한다.

NASA GSFC에서 공개한 cFE(core Flight Executive)는 지상에서 업로드 된(또는 EEPROM 또는 SGM에 저장된) 파일에서 소프트웨어 설정 정보를 읽어드려 처리함으로써 위성 온-보드 시스템에 MS 윈도우의 레지스트리 개념을 도입하였다. 그리고 소프트웨어 모듈 로드 스크립트(load script)를 통해서 처음 시스템이 부팅할 때 EEPROM/SGM에 보관된 시스템 소프트웨어 모듈 라이브러리 중에서 원하는 것만 로드할 수 있게 하였다(그러나 이것은 파일을 기반으로 했을 때만 가능한 것은 아님).

소프트웨어 모듈이 파일로 저장되어 있다는 것은 선택적으로 다른 저장소에서 로드하는 것이 가능하게 하는 것이며 소프트웨어 재프로그래밍(reprogramming)을 보다 쉽게 한다. 예를 들어 EEPROM 또는 SGM에 저장된 항법유도 제어 소프트웨어 모듈을 /EEPROM/gnc1.o 또는 /SGM/gnc2.o에서 초기에 읽어 들였다면 TC로 새로운 소프트웨어 모듈을 다운 받은 후 /TC/gnc\_new.o를 다음 부팅때에 읽어들여 새로운 소프트웨어 모듈을 로드할 수 있다.

#### 4.4.2 운영 파라미터 관리

SGM은 특히 파일 시스템을 통한 위성 운영에 유용하다. 그 이유는 SGM은 컴퓨터의 리셋에 영향을 받지 않기 때문에 컴퓨터 운영 중 또는 리셋 후에 파일의 내용을 적용하는 것이 가능하기 때문이다. SGM에는 MTL (Mission Time Line) 또는 탑재체 운용 상태, 동작 모드, 에러 리포트, 그리고 소프트웨어 패치 등을 파일로 보관가능하다<sup>(8)</sup>. 파일 시스템을 통해서 ICD를 계속 업데이트 하지 않고도 해당 미션 파라미터 파일에 접근하여 위성 운영에 이 정보를 반영할 수 있다.

또한 ESA의 경우 OBCP (On-Board Control Procedure)의 개발 및 적용이 확대되어 가고 있는 상황인데 OBCP 코드 로드 및 처리도 파일로 수행될 수 있다<sup>(8, 10)</sup>.

크기 등 컴퓨터 설계에 대해서 고민해야 할 사항이 줄어들

- efficiency
  - : 패킷 헤더를 따로 저장할 필요가 없어 공간 활용이 효율적이고 온-보드 파일에 대해서 일반 압축 알고리즘 적용 가능<sup>(8)</sup>

파일 시스템을 사용했을 때 예상되는 문제점은 파일 시스템 라이브러리의 용량만큼 전체 소프트웨어 이미지 크기가 증가한다는 점과, 파일 시스템을 사용하기 위해서 프로세서 성능을 추가적으로 요구한다는 점이다. 따라서 파일 시스템을 적용할 때 온-보드에 탑재된 EEPROM 사이즈 또는 프로세서 처리량(throughput), 전체 소프트웨어 이미지 증가에 따른 재프로그램밍 영향과 같은 운용적인 측면이 고려되어야 할 것이다.

### 5. FMS 운용시 장단점

파일 시스템을 채택했을 때 얻을 수 있는 가장 큰 이점은 아래와 같은 높은 수준의 자동화이다.

- automatic data delivery and control
  - : 날짜별, 이름별 정렬을 통해 원하는 데이터 선택 가능
  - : 조건에 따라서 데이터의 자동 재전송, 이어받기, 또는 삭제 가능
  - : 테이블 로드 (table load)가 용이해 짐
  - : 임의 형태(크기, 포맷 등)의 파일 생성이 용이해 짐
  - : 온-보드 소프트웨어 개발시 NFS (Network File System) 등과 같은 네트워크 파일 기반 개발 기술(TFTP, FTP 등) 활용 가능
  - : RAM에 램 디스크 (RAM disk)를 생성하여 중간 단계의 데이터를 처리한 후 최종에 저장 장치로 이동 용이
- minimized interconnection to computer design
  - : 파일/디렉토리의 생성, 변경, 삭제 등을 FMS이 자동으로 해 주므로 유저가 RAM

### 6. 결 론

위성을 비롯한 우주 탐사선에서 파일 시스템의 사용은 그 동안 큰 저장 장치를 사용하고 좀 더 독립적이고 단위 처리를 필요로 했으나 패킷 단위 통신 시스템을 사용했던 분야에서 이점을 발휘할 수 있다.

패킷 통신 시스템과 파일 기반 통신 시스템의 장단점을 분석하여 시스템의 성격에 맞는 방식을 채택하면 보다 최적화된 위성 운영을 할 수 있리라 판단된다. 특히 우주 탐사선과 같이 지구에서 멀리 떨어져 통신 신호 세기의 감소 및 교란이 예상됨에도 불구하고 제한된 전력 및 자원 하에서 복잡한 미션, 많은 탑재체, 심각하게 자동화 처리가 요구되는 경우에 있어 파일 시스템의 사용이 보다 많은 이점을 주는 것으로 판단된다.

파일에 기반을 둔 온-보드 처리는 DTN 통신의 기본 요구사항이므로 CFDP나 BP와 같은 DTN 프로토콜과 온-보드 파일 시스템과의 접속 아키텍처를 기술하고 있는 CCSDS SOIS를 채택한 온-보드 소프트웨어 구성에 대한 연구도 진행하면 좋을 것이다.

지금까지 본 논문에서 우주 탐사선에 적용되고

있는 파일 시스템의 현황 및 그 응용 기술을 분석한 결과를 수록하였고 추후에는 이와 연계된 CFDP와 같은 DTN 프로토콜에 대한 연구가 진행될 것이다.

## 참 고 문 헌

1. [http://en.wikipedia.org/wiki/File\\_system](http://en.wikipedia.org/wiki/File_system)
2. Joerg Mueller, "Solid State Mass Memory with integrated File Handling System for Mars Express and Venus Express", 2003, DASIA 2003
3. ECSS-E-70-41A, "Ground systems and operations – Telemetry and telecommand packet utilization", 2003, ESA ECSS
4. CCSDS 727.0-B-4, "CCSDS File Delivery Protocol (CFDP)", 2007, CCSDS
5. CCSDS 211.0-B-4, "Proximity-1 Space Link Protocol-DATA LINK LAYER", 2006, CCSDS
6. CCSDS 873.0-M-1, "Spacecraft Onboard Interface Services-File and Packet Store Services", 2012, CCSDS
7. M. Pecchioli, "Proposed concepts for file based operations", 2012, ADCSS 2012
8. Brice Dellandrea et al, "Files in space : management & transfer", 2013, DASIA 2013
9. Riaz Shafi et al, "File based operations : The future of satellite operations", 2013, EGOS
10. Rick Blake, "Deployment of file based spacecraft communication protocols", 2012, SpaceOps 2012
11. Jonathan Wilmot, "Use of CCSDS File Delivery Protocol (CFDP) in NASA/GSFC's Flight Software Architecture: core Flight Executive (cFE) and Core Flight System (CFS)", 2012, ADCSS 2012