

달 착륙선 데이터 처리시스템 연구

석병석*

A study on the on-board data handling system for lunar lander

Suk, Byong-Suk*

ABSTRACT

The KPLO (Korea Path-finder Lunar Orbiter) project was started in the beginning of 2016 and it was the first phase of national lunar exploration plan. Sooner or later, the second phase lunar exploration project will be started with lunar landing mission. In this paper, the on-board data handling(OBDH) system of lunar lander was analyzed based on the several case studies. Each OBDH system architectures for pin-point landing was shown and major design considerations are summarized. The major difference of OBDH system between earth observation satellite and lunar lander is main computer architecture. To increase reliable and powerful processing than earth observation satellite, it is trending for lunar lander to adopt distributed computer architecture to provide fail safe redundancy and required throughput.

초 록

현재 우리나라는 시험용 달 궤도선 개발을 시작으로 달 탐사 1단계 사업을 착수하였으며, 조만간 달 탐사 2단계 사업에서는 달착륙선 개발을 계획하고 있다. 본 논문은 향후 개발될 달착륙선의 온 보드 데이터 처리 시스템에 대한 개념 연구를 수행하였으며 데이터 처리 시스템 설계를 위한 주요 요구조건에 대한 해외사례를 분석하였다. 지구 저궤도 위성의 경우 대부분 중앙 집중 방식의 단일 프로세서를 가지는 온보드 데이터 처리 시스템 설계가 일반적인 반면 달 착륙선의 경우 착륙의 신뢰도를 높이기 위해 이중 프로세스 구조의 온보드 데이터 처리 시스템으로 설계하는 추세이다. 이러한 이중 프로세스 구조는 착륙을 위한 영상항법에 필요한 프로세스 연산 속도를 높이고 프로세스 고장 발생에 대한 신뢰성 향상에 적합한 구조이다.

Key Words : 달 착륙선(Lunar Lander), 데이터 처리 시스템(On-Board Data Handling)

* 석병석, 한국항공우주연구원, 위성연구본부 달탐사선행입무팀
byongss@kari.re.kr

1. 서론

우리나라는 달 탐사 1단계로 시험용 달 궤도선 개발 사업을 2016년부터 시작하였다. 조만간 2단계 사업으로 달착륙선 개발 사업이 시작될 예정이다. 본 논문은 달착륙선 온보드 데이터 처리 시스템의 주요 요구조건들을 분석하기 위해 해외 달착륙선 데이터 처리 시스템의 사례를 분석하고 연구하였다.

최근 선진국들은 달 탐사에 다시 관심을 가지기 시작했으며 특히 달의 특정 지역에 착륙하는 pin-point landing 시스템 개발을 추진하고 있다[1]. 원하는 지점에 안전하고 정확하게 착륙하기 위해서는 온보드 데이터 처리 시스템이 가장 중요한 역할을 담당하게 되며 특히 카메라 및 레이더 등을 사용하는 착륙 하강 단계에서는 데이터 처리 시스템이 더욱 중요한 역할을 수행하게 된다. 일반적으로 지구저궤도 위성의 경우 중앙 집중 형태의 단일 프로세스 구조를 대부분 사용하고 있다. 그러나 최근 달 착륙선의 경우 영상항법을 위한 빠른 연산 속도가 요구되며 또한 착륙과정에서 프로세서 오류 발생에 대한 대처를 위해 프로세서 이중화 구조를 많이 채택하고 있다.

2. 데이터 처리 시스템 구조

본 논문에서는 달 뿐만 아니라 화성 착륙선의 데이터 처리 시스템도 포함하여 사례연구를 수행하였다. 화성 착륙선과 달착륙선 설계에는 많은 유사점이 있으며 다만 화성은 대기가 존재하지만 달은 대기가 없으므로 달 착륙이 화성 착륙에 비해 더 도전적인 것으로 알려져 있다 [1].

2.1 설계 고려사항

초기 행성 착륙 임무에서 가장 중요하게 생각한 목표는 구조물의 안정성과 기능적인 운용

성 확보에 주력하였으며, 이러한 목표 달성을 위해 첫째 인간이 제어루프에 포함되는 경우 (아폴로 임무), 둘째 에어백을 이용한 착륙, 세 번째 레이더 고도계를 이용한 반자동 착륙 방식 등이 고려되었다. 그러나 최근에는 과거와 달리 실시간으로 위험을 감지하고 회피하면서 안전한 착륙을 시도하게 되었으며 특히 저비용으로 빠르게 데이터 처리 시스템을 개발하는 전략이 하나의 경향으로 자리 잡고 있다[1]. 또한 과거에는 중앙 집중 형태의 데이터 처리 시스템으로 모든 요구조건을 만족할 수 있었지만 최근에는 ExoMars 및 창어1호와 같이 분산 구조 형태의 데이터 처리 시스템을 적용하는 추세이다[1]. 만약 실시간으로 위험을 감지하고 회피하면서 pin-pointing 착륙이 요구된다면 이를 위해 필요한 일반적인 데이터 처리 시스템의 요구조건을 <표 1>에 나타내었으며 우리나라 착륙선 개념설계에 참고할 필요가 있다.

표 1. Overall processing power & memory requirements for pin-pointing landing

Task	Throughput	Code Memory	Data Memory
Operating System	-	4M	8M
Mass Memory	-	-	64M
LIDAR	3-5 MIPS	1-1.5 MB	16 MB
Descent Camera	1.5-2.5 MIPS	0.5-1 MB	16 MB
Other tasks	5-6 MIPS	988 KB	600 KB
Autonomy	0.5-1 MIPS	05 MB	2 MB
Hazard detection & avoidance	0.5-1 MIPS	0.5 MB	2 MB
Total	Approx. 15.5 MIPS	7.5 MB rounded to 8 MB	108 MB rounded to 128 MB

착륙선 데이터 처리 시스템 설계에서 또 하나 중요한 고려 사항은 에러 발생에 대한 안전성 확보와 이중화(fail-safe/fail-operation)이다. 특히 짧은 시간이긴 하지만 착륙 하강 단계에서 만약 프로세스 에러가 발생할 경우 임무 실패로 연결될 수 있는 심각한 상황이 전개될 수 있다. 따라서 primary 컴퓨터에서 오류가 발생할 경우 redundant 컴퓨터가 수행 중인 모든 task를 중단이나 지연 없이 곧바로 연속하여 수행할 수 있는 기능이 달 착륙선에는 요구된다.

2.2 해외 착륙선 사례

2.2.1 Viking Lander

Viking은 화성탐사 프로젝트로 1975년 7월에 화성에 착륙한 바 있다. <그림 1>의 Viking lander의 컴퓨터 아키텍처에서 알 수 있듯이 중앙 집중(centralized) 구조이며 항법제어(Guidance & Control)부분과 데이터 처리(DAPU: Data Acquisition & Processing Unit) 두 부분으로 나누어져 있음을 알 수 있다. Viking 프로젝트를 통하여 밝혀진 중요한 사실은 착륙을 위한 완벽한 제어루프 수행을 위해서는 20ms의 시간이 소요된다는 것이며 이 사실은 이후 pin-point landing을 위한 착륙선 컴퓨터 시스템 설계에 참고하고 있다[1].

2.2.2 Mars Pathfinder

Mars pathfinder는 화성의 대기 기후 지질 등을 탐사하기 위해 1997년 7월 발사되었다. <그림 2>는 mars pathfinder의 데이터 처리 시스템을 나타내는 그림으로 AIM (Attitude and Information Management)을 중심으로 하는 중앙 집중 구조로 설계되었다. 모든 유닛들은 1553B 및 VME 버스를 통하여 연결되는 구조이다. 이러한 하이브리드 구조를 설계한 이유는 고속의 데이터 처리는 VME 버스를 통해 이루어지도록 하고 저속의 데이터 처리는 155B

버스를 통해 처리하기 위함이다. 이중화 버스 구조로 설계되어 있지만 프로세서의 고장에 대해서는 대안이 없는 single point failure 가능성이 있으며 만약 이러한 고장의 경우 임무 전체 실패로 이어지는 구조이다.

2.2.3 ExoMars

<그림 3>은 ExoMars(Exobiology on Mars)의 데이터 처리 시스템 구조를 나타낸다. 그림에서처럼 분산처리(distributed) 구조로 두 개의 CDMU(Control & Data Management Unit), 탑재체 RTU(Remote Terminal Unit), engineering RTU, Attitude Information Unit(AIU)로 구성되어 있다. 그러나 이러한 구조는 고장 복구가 가능한 fail-safe 이중화 구조이긴 하지만 데이터 버스 속도가 500kbps로 느리고 메모리 액세스가 1553B 버스로 이루어져서 속도가 느리기 때문에 착륙미션에 적합하지 않은 것으로 알려져 있다[1].

2.2.4 창어-1

창어1호는 중국의 첫 번째 달 궤도선으로 <그림 4>에 나타난 바와 같이 데이터 처리 시스템은 분산처리(distributed) 구조이며, 1553B 버스를 이용하여 데이터를 주고받는 구조이다. 고장 발생을 대비하여 각각의 모듈을 이중화하여 설계하였다. 창어 1호 임무는 착륙선이 아닌 궤도선의 경우로 착륙 하강단계와 같은 아주 중요한 데이터 처리 기능은 필요 없기 때문에 1553B 버스로만 구성한 것으로 생각된다.

2.2.5 기술 동향

앞서 나타난 것처럼 초기 행성 탐사인 viking 과 mars pathfinder의 경우 데이터 처리 시스템의 구조는 중앙 집중 식(centralized) 구조로 설계가 되었으나 점차로 기술이 발전하여 이후에 등장하는 ExoMars와 창어-1호의 경

우 분산처리(distributed) 구조로 가고 있으며 또한 오류에 대비한 프로세서 이중화 구조로 설계하는 추세를 보이고 있다.

최근 컴퓨터 및 반도체 집적기술의 비약적인 발전으로 매우 빠른 데이터 처리가 가능하며 단일 프로세서로 주어진 임무를 모두 수행하는 것이 가능하지만 그러나 만약 메인 프로세서에서 예기치 못한 오류가 발생할 경우를 대비하여 프로세서 이중화 설계가 최근 추세이다. 이러한 프로세스 이중화 요구와 더불어 착륙선의 자세제어 알고리즘을 수행하는 컴퓨터와 카메라 영상항법 처리 컴퓨터를 분리하는 추세이다.

3. 결론

달착륙선 데이터 처리 시스템에 대한 해외 사례를 조사 분석하였으며, 행성탐사 초기 착륙 임무 수행을 위한 데이터 처리 시스템은 중앙 집중 형태의 구조가 많이 사용되었으나 최근에는 실시간 정밀 영상항법 및 정밀 착륙을 위해 분산 구조로 설계하고 있다. 또한 사례에서도 보듯이 착륙 하강 단계에서의 고장 발생

은 임무 실패로 이어지는 심각한 상황이 발생할 수 있으므로 이에 대비한 이중화 설계가 필수적임을 알 수 있다.

참고문헌

1. Tanya Vladimirova, Muhammad Fayyaz, Martin N. Sweeting, Valentin I. Vitanov "A novel autonomous low-cost on-board data handling architecture for a pin-point planetary lander," ELSEVIER, Acta Astronautica 68(2011), pp. 811-829, Oct 2010.
2. Barbara A. Cohen, D. Gregory Chavers, Benjamin W. Ballard, "NASA's Robotic Lunar Lander Development Project," ELSEVIER, Acta Astronautica, 79(2012), pp. 221-240, May 2012

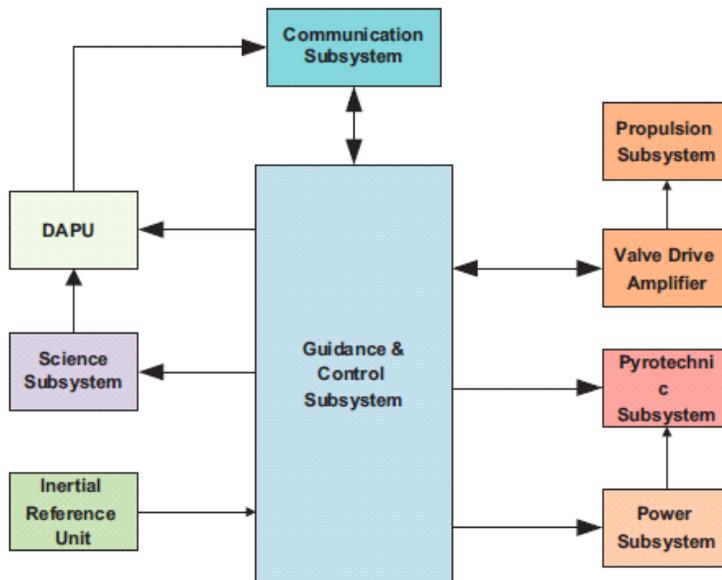


그림 1. Functional Block Diagram of the Viking lander computer system[1]

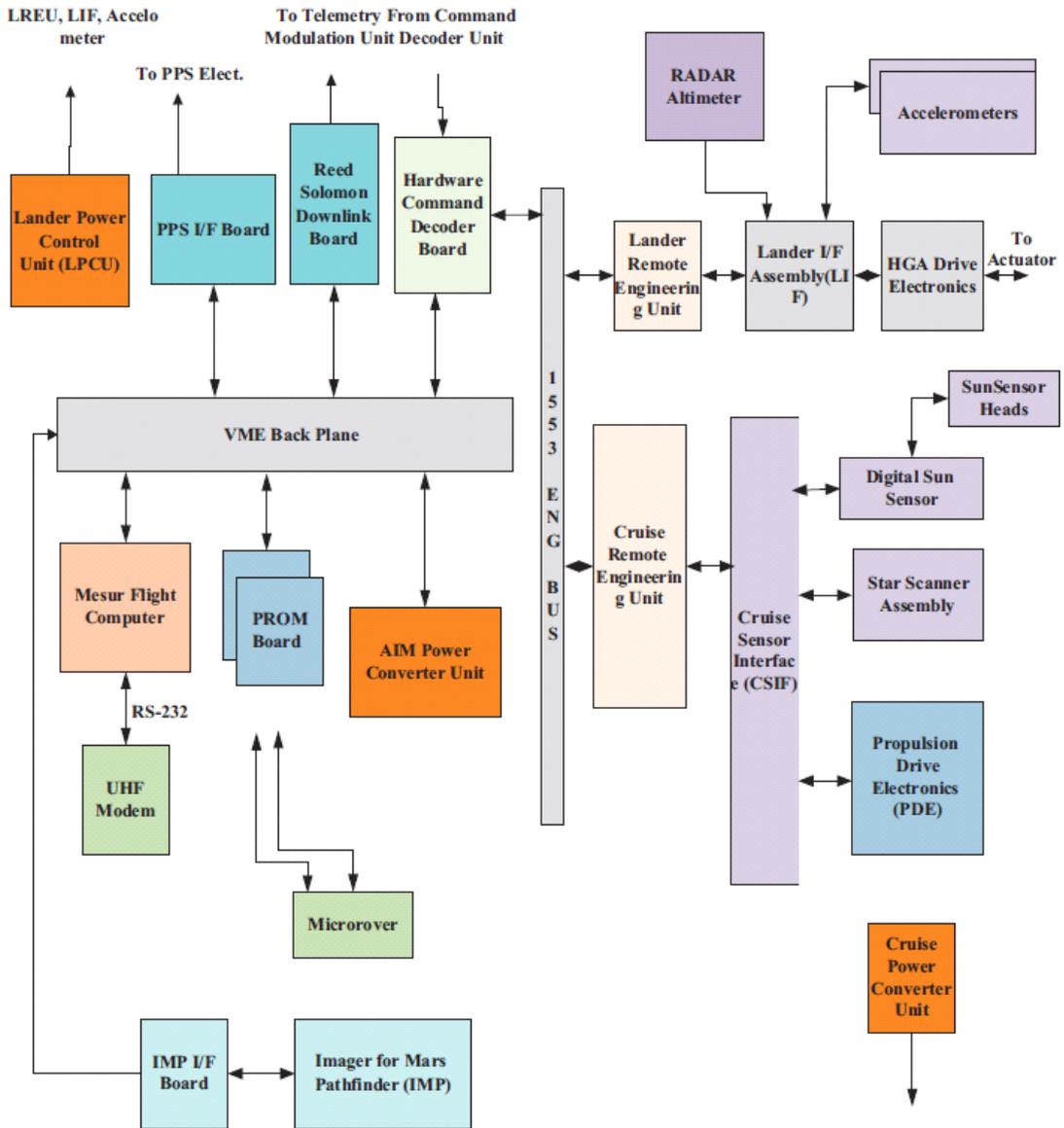


그림 2. Mars Pathfinder Attitude and Information Management Unit[1]

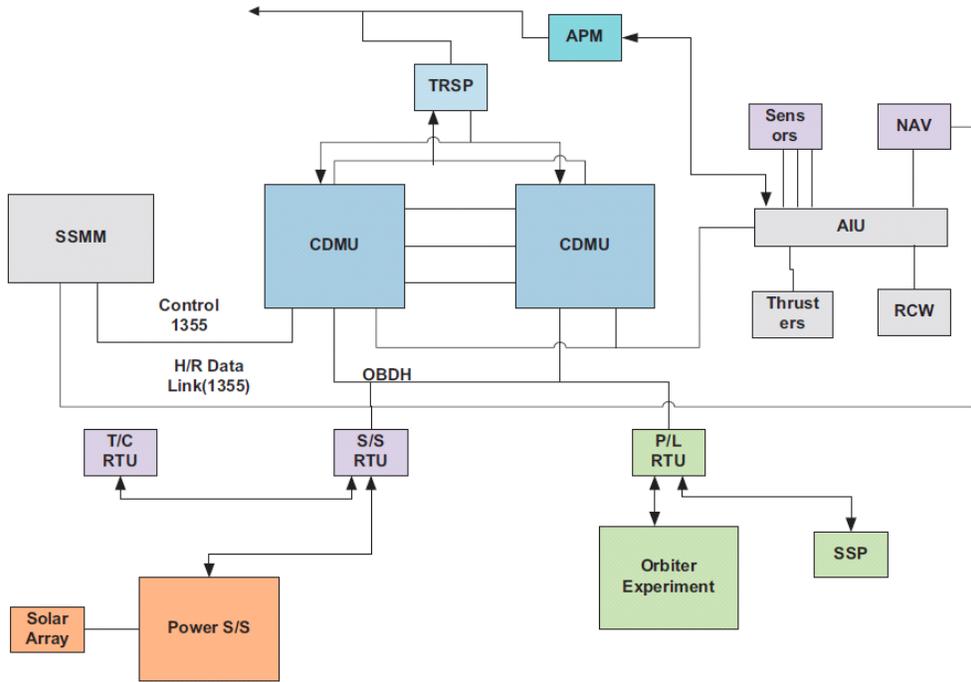


그림 3. Block Diagram of the ExoMars09 on-board data handling system[1]

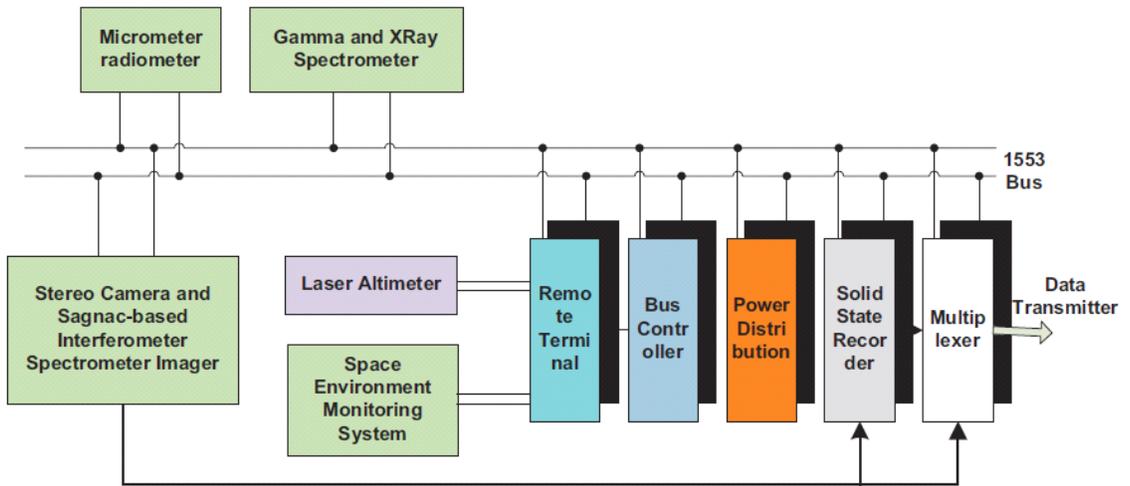


그림 4. Chang's E-1 on-board data handling system[1]