

|                  |   |
|------------------|---|
| Title            | Satellite data handling system design for architectural-layer-driven verification   |
| Sub Title        |   |
| Author           | Ngo, Thi Hoai(Shirasaka, Seiko)<br>白坂, 成功   |
| Publisher        | 慶應義塾大学大学院システムデザイン・マネジメント研究科   |
| Publication year | 2016  |
| Jtitle           |   |
| JaLC DOI         |   |
| Abstract         |   |
| Notes            | 修士学位論文. 2016年度システムエンジニアリング学 第223号   |
| Genre            | Thesis or Dissertation  |
| URL              | <a href="https://koara.lib.keio.ac.jp/xoonips/modules/xoonips/detail.php?koara_id=KO40002001-00002016-0006">https://koara.lib.keio.ac.jp/xoonips/modules/xoonips/detail.php?koara_id=KO40002001-00002016-0006</a> |

慶應義塾大学学術情報リポジトリ(KOARA)に掲載されているコンテンツの著作権は、それぞれの著作者、学会または出版社/発行者に帰属し、その権利は著作権法によって保護されています。引用にあたっては、著作権法を遵守してご利用ください。

The copyrights of content available on the KeiO Associated Repository of Academic resources (KOARA) belong to the respective authors, academic societies, or publishers/issuers, and these rights are protected by the Japanese Copyright Act. When quoting the content, please follow the Japanese copyright act.

# Satellite Data Handling System Design for Architectural-Layer-Driven Verification

NGO THI HOAI

(Student ID Number : 81434568)

Supervisor Prof. Seiko Shirasaka

September 2016

Graduate School of System Design and Management,  
Keio University  
Major in System Design and Management

## SUMMARY OF MASTER'S DISSERTATION

|   |          |      |              |
|---|----------|------|--------------|
| Student<br>Identification<br>Number   | 81434568 | Name | Ngo Thi Hoai |
| Title<br><br>Satellite Data Handling System Design for Architectural-Layer-Driven Verification  |          |      |              |
| Abstract<br><p>Satellite Data Handling System (DHS) is a data exchange system between flight segment (satellite) and ground segment. The satellite and ground segment are often independently-developed so that confirmation of mutual understanding between those two segments is a challenge. However, it is an essential step, as it is needed to verify whether the satellite in orbit has ability to communicate with the ground segment on ground. DHS interfaces are critical. It may be easy to confirm physical interfaces, however, confirmation of logical interfaces is not. The risk of that those interfaces do not work well, is not small, especially, for the case of a new DHS system development. To reduce the risk, those interfaces should be confirmed as soon as possible. If this confirmation is done when the integration of two segments are completed, and the result shows that those interfaces have problems, it will take a lot of steps back and forth to identify and fix the problems.</p> <p>For decades, a lot of approaches have been developed and introduced for testing DHS interfaces between satellite and ground segment. Some approaches have offered great ways to design and verify DHS. However, these approaches still miss to introduce systematic methodology to identify how upper layers of DHS interfaces can be verified at low-level testing, before system-level tests.</p> <p>In this research, I am about to propose a new approach to satellite data handling system design for architectural-layer-driven verification. In this approach, processes of architecture design and verification of data handling system are to be introduced in systematic way. DHS architecture is categorized into two types according to different viewpoints of design. One of two types, “vertical” architecture, is designed to illustrate upper layer interfaces, which cannot be shown in the other type, “horizontal” architecture. This way of architecture design supports verification of DHS upper-layers interfaces at low-level testing, thus, objective of the research, reduction of</p> |          |      |              |

development risks and iteration testing processes. Before experienced engineers have been trying to test these interfaces based on their experience without any explicit systematic way. Evaluation of the approach is to be done by applying the approach to MicroDragon satellite developed by Vietnam National Satellite Center (VNSC), and to a data relay satellite system, and by interviewing five satellite specialists.

#### Key Words

Satellite Data Handling System, MicroDragon satellite, Satellite test, DoDAF, etc.

## **Acknowledgements**

Many people have helped me to complete the thesis. Firstly, I would like to thank my supervisor, Prof. Seiko Shirasaka, for spending time on helping me and giving me valuable comments and advices, and for his great support during the time I've studied in Keio University. My honest thanks go to professors at Keio SDM and Tokyo University including Prof. Makoto Ioki, Ms. Asako Kageki, Prof. Takashi Hiramatsu, Prof. Shusaku Yamaura, Prof. Ayumu Tokaji, Prof. Shinichi Nakasuka, and Prof. Yoshihiro Tsuruda who have shared experiences, and given me valuable insights, supports, and essential pushes. I would like to thank my second supervisor, Prof. Tetsuya Toma, for his great advices and comments.

I would like to thank my colleagues at Vietnam National Satellite Center for giving me spiritual and physical supports all the time I've stayed in Japan. My sincere appreciation goes to Vietnam National Satellite Center that has given me a great opportunity, and provided financial support for my study.

Finally, my special thanks go to my husband, and my family members, for giving me their endless love, and always being by my side through thick and thin.



# Table of Contents

|   |     |
|---|-----|
| Acknowledgements.....   | III |
| Table of Contents.....  | V   |
| List of tables.....   | VI  |
| List of figures.....  | VII |
| Abbreviation .....  | X   |
| Chapter 1    Introduction .....   | 1   |
| 1.1. Research Background.....   | 1   |
| 1.1.1. Overview of satellite and ground segment.....  | 1   |
| 1.1.2. Introduction to satellite Data Handling System .....   | 5   |
| 1.1.3. Introduction to MicroDragon satellite .....  | 9   |
| 1.1.4. Introduction to TableSat .....   | 13  |
| 1.2. Research Objective.....  | 15  |
| 1.3. Scope of Research .....  | 16  |
| 1.4. Outline of dissertation .....  | 16  |
| Chapter 2    Literature Review .....  | 18  |
| 2.1. Survey of existing approaches to satellite development.....  | 18  |
| 2.1.1. Model-based development and verification (MDV).....  | 18  |
| 2.1.2. “Test like you fly” approach.....  | 21  |
| 2.1.3. Orbital Sciences spacecraft integration and test suite based on “test as you fly”<br>approach.....                           | 26  |
| 2.1.4. Best practices for verification, validation, and test in model-based design...   | 30  |
| 2.2. Discussion and conclusion on the existing approaches.....  | 32  |
| Chapter 3    Proposal:    Satellite    Data    Handling    System    Design    for<br>Architectural-Layer-Driven Verification ..... | 34  |
| 3.1. Introduction to the approach to satellite data handling system design for<br>architectural-layer-driven verification.....      | 34  |
| 3.1.1. Goal of the approach.....  | 34  |
| 3.1.2. Detailed approach .....  | 34  |
| 3.1.3. Introduction to Department of Defense Architecture Framework (DoDAF)   | 42  |
| 3.2. Originality of the approach.....   | 49  |

|                 |  |     |
|-----------------|--|-----|
| Chapter 4       | Evaluation of the approach.....  | 51  |
| 4.1.            | Evaluation of the approach.....  | 51  |
| 4.1.1.          | Applying the approach to develop DHS of MicroDragon satellite.....                     | 51  |
| 4.1.2.          | Applying the approach to develop DHS of a data relay satellite system.....             | 51  |
| 4.1.3.          | Interviewing satellite specialists .....   | 51  |
| 4.2.            | Applying the approach to develop DHS of MicroDragon satellite .....                    | 57  |
| 4.2.1.          | Architecture design of DHS of MicroDragon satellite .....                              | 57  |
| 4.2.2.          | Verification of MicroDragon DHS .....  | 82  |
| 4.3.            | Applying the approach to develop DHS of the data relay satellite system .....          | 87  |
| 4.4.            | Interview with satellite specialists.....  | 91  |
| 4.4.1.          | Interviewee No. 1 .....  | 91  |
| 4.4.2.          | Interviewee No.2.....  | 97  |
| 4.4.3.          | Interviewee No.3 .....   | 100 |
| 4.4.4.          | Interviewee No.4.....  | 105 |
| 4.4.5.          | Interviewee No.5 .....   | 108 |
| 4.5.            | Discussion and conclusion on results of evaluation of the approach .....               | 111 |
| 4.5.1.          | Results of applying the approach to develop DHS of MicroDragon satellite .....         | 111 |
| 4.5.2.          | Results of applying the approach to develop DHS of a data relay satellite system ..... | 112 |
| 4.5.3.          | Results of interviewing satellite specialists .....                                    | 112 |
| Chapter 5       | Conclusion.....  | 114 |
| 5.1.            | Summary .....  | 114 |
| 5.2.            | Future work .....  | 114 |
| References..... |  | 115 |

## List of tables

|          |   |    |
|----------|---|----|
| Table 1. | List of command types.....                          | 7  |
| Table 2. | Distinction between TLYF and other tests .....      | 25 |
| Table 3. | Heater test data elements <sup>[3]</sup> .....      | 29 |
| Table 4. | Operational view in DoDAF 1.0 <sup>[24]</sup> ..... | 43 |

|   |     |
|---|-----|
| Table 5. Systems view in DoDAF 1.0 <sup>[24]</sup> .....                          | 47  |
| Table 6. List of interviewees for evaluation of the approach .....                | 52  |
| Table 7. List of questions for evaluation of the approach.....                    | 53  |
| Table 8. Functional allocation of DHS of MicroDragon satellite.....               | 78  |
| Table 9. Mapping table between operational activities and systems functions ..... | 79  |
| Table 10. Experiment procedures in one TableSat activity .....                    | 84  |
| Table 11. Interview Results with the interviewee No. 1 .....                      | 91  |
| Table 12. Interview Results with the interviewee No. 2.....                       | 97  |
| Table 13. Interview Results with the interviewee No. 3 .....                      | 100 |
| Table 14. Interview results with the interviewee No. 4 .....                      | 105 |
| Table 15. Interview results with the interviewee No. 5 .....                      | 108 |

## List of figures

|   |    |
|---|----|
| Figure 1. Satellites of NASA to study the oceans, land, and atmosphere [credits: NASA]<br>..... | 1  |
| Figure 2. Common subsystems of a satellite .....  | 2  |
| Figure 3. Number of Spacecraft launched from 2005 to 2014 <sup>[2]</sup> .....                  | 3  |
| Figure 4. An example of ground segment.....   | 3  |
| Figure 5. International Space Station control room in Russia .....                              | 4  |
| Figure 6. Ground station of Galileo satellites in Redu, Belgium.....                            | 5  |
| Figure 7. An example of Data Handling System between satellite and ground segment.              | 6  |
| Figure 8. An example of a taking-picture block command .....                                    | 7  |
| Figure 9. Execution of a timeline command.....  | 8  |
| Figure 10. Lifecycle of satellite data handling system .....                                    | 9  |
| Figure 11. 3D-Image of MicroDragon satellite .....  | 10 |
| Figure 12. Nadir Scan Mode of MicroDragon satellite .....                                       | 11 |
| Figure 13. Tilt Scan Mode of MicroDragon satellite.....   | 12 |
| Figure 14. Target Scan Mode of MicroDragon satellite .....                                      | 12 |
| Figure 15. Coastal Scan Mode of MicroDragon satellite .....                                     | 13 |
| Figure 16. Conceptual scenario of secondary mission of MicroDragon satellite .....              | 13 |
| Figure 17. TableSat layout at Hongo campus, Tokyo University.....                               | 14 |

|   |    |
|---|----|
| Figure 18. TableSat satellite room at Tokyo University .....  | 14 |
| Figure 19. TableSat data acquisition system at Tokyo University .....   | 15 |
| Figure 20. Development V-Model of Data Handling System .....  | 16 |
| Figure 21. Satellite development and operation phases defined in MDV <sup>[7]</sup> .....   | 18 |
| Figure 22. MDVE configuration <sup>[7]</sup> .....  | 19 |
| Figure 23. MDVE configuration in phase B <sup>[7]</sup> .....   | 20 |
| Figure 24. MDVE configuration in phase C <sup>[7]</sup> .....   | 20 |
| Figure 25. MDVE configuration for phase D <sup>[7]</sup> .....  | 21 |
| Figure 26. MDVE for phase E <sup>[7]</sup> .....  | 21 |
| Figure 27. TLYF process to access mission concepts for testability and to access risk <sup>[25]</sup><br>.....                        | 23 |
| Figure 28. TLYF operability test technique <sup>[25]</sup> .....  | 24 |
| Figure 29. TLYF total operations chain test <sup>[25]</sup> .....   | 24 |
| Figure 30. Key to injection of “like you fly” tests <sup>[25]</sup> .....   | 25 |
| Figure 31. “Telemetry point retrieval logic” algorithm <sup>[3]</sup> .....   | 28 |
| Figure 32. Thermal Control System Health Test Sequence <sup>[3]</sup> .....   | 30 |
| Figure 33. Test suites running against MIL, SIL, and PIL <sup>[22]</sup> .....  | 31 |
| Figure 34. Introduction to the approach to satellite data handling system design for<br>architectural-layer-driven verification ..... | 35 |
| Figure 35. Basis of the approach.....   | 35 |
| Figure 36. Relationship between operation architecture domain and systems architecture<br>domain.....                                 | 36 |
| Figure 37. Open System Interconnection Model .....  | 37 |
| Figure 38. An example of vertical architecture of a satellite data handling system .....  | 38 |
| Figure 39. Verification of Telemetry/ Command .....   | 39 |
| Figure 40. Verification of Mission Data .....   | 40 |
| Figure 41. Verification of RF link by coaxial cable connection.....   | 41 |
| Figure 42. End-to-end testing configuration for verification of RF link .....   | 41 |
| Figure 43. Operational activity hierarchy chart and operation activity diagram <sup>[24]</sup> .....                                  | 45 |
| Figure 44. Decomposition model of systems functionality description (SV-4) .....  | 48 |
| Figure 45. System data flow model of systems functionality description (SV-4) .....   | 48 |
| Figure 46. Operational activity to systems function traceability matrix (SV-5).....   | 49 |
| Figure 47. A data relay satellite system .....  | 50 |

|   |    |
|---|----|
| Figure 48. Concept of operations of MicroDragon satellite .....   | 57 |
| Figure 49. High-level operational activities of DHS of MicroDragon satellite.....                           | 58 |
| Figure 50. High-level operational activities flow of DHS of MicroDragon satellite .....                     | 59 |
| Figure 51. Operational activities in initial operations of DHS of MicroDragon satellite<br>(A1).....        | 60 |
| Figure 52. A1 operational activities flow of DHS of MicroDragon satellite.....                              | 61 |
| Figure 53. Operational activities in nominal phase of DHS of MicroDragon satellite<br>(A2).....             | 62 |
| Figure 54. A2 operational activities flow of DHS of MicroDragon satellite.....                              | 63 |
| Figure 55. Operational activities in end-of-operations of DHS of MicroDragon (A3) ..                        | 63 |
| Figure 56. A3 operational activities flow of DHS of MicroDragon satellite.....                              | 64 |
| Figure 57. Operational nodes of DHS of MicroDragon satellite .....  | 65 |
| Figure 58. Operational activities assignment to operational nodes in DHS of<br>MicroDragon satellite .....  | 65 |
| Figure 59. Context diagram of DHS of MicroDragon in initial operations phase.....                           | 67 |
| Figure 60. Use case diagram of DHS in initial operations phase of MicroDragon<br>satellite .....            | 68 |
| Figure 61. Functional requirements of DHS in initial operations phase of MicroDragon<br>satellite .....     | 69 |
| Figure 62. Context diagram of DHS in nominal operations phase of MicroDragon<br>satellite .....             | 70 |
| Figure 63. Use case diagram of DHS in nominal operations phase of MicroDragon<br>satellite .....            | 71 |
| Figure 64. Functional requirements (1) of DHS in nominal operations phase of<br>MicroDragon satellite ..... | 72 |
| Figure 65. Functional requirements (2) of DHS in nominal operations phase of<br>MicroDragon satellite ..... | 73 |
| Figure 66. Context diagram of DHS in end-of-operations phase of MicroDragon<br>satellite .....              | 73 |
| Figure 67. Use case diagram of DHS in end-of-operations phase of MicroDragon<br>satellite .....             | 74 |
| Figure 68. Functional requirements of DHS in end-of-operations phase of MicroDragon<br>satellite .....      | 74 |

|   |           |
|---|-----------|
| Figure 69. Function flow block diagram of DHS of MicroDragon satellite.....                           | 75        |
| Figure 70. Sequence of temperature control in initial operations phase of MicroDragon satellite ..... | 75        |
| Figure 71. Sequence of supplying electric power in nominal phase of MicroDragon satellite .....       | 76        |
| Figure 72. Sequence of TLM/ CMD transmission in nominal phase of MicroDragon satellite .....          | 76        |
| Figure 73. Sequence of performing mission in nominal phase of MicroDragon satellite .....             | 77        |
| Figure 74. Physical design of DHS of MicroDragon satellite .....                                      | 78        |
| Figure 75. Horizontal architecture of DHS of MicroDragon satellite .....                              | 81        |
| Figure 76. Vertical architecture of DHS of MicroDragon satellite .....                                | 82        |
| Figure 77. Testing TLM/ CMD layout with TableSat in Tokyo University .....                            | 83        |
| Figure 78. Sending CMD from DAS computer to MOBC in one TableSat activity .....                       | 84        |
| Figure 79. Checking LEDs status in checkout equipment in one TableSat activity .....                  | 86        |
| Figure 80. Connecting components with cables in one TableSat activity .....                           | 86        |
| Figure 81. Monitoring TLM in one TableSat activity .....  | 87        |
| Figure 82. Architecture of DHS of the data relay satellite system .....                               | 89        |
| <b>Figure 84. Verification of data link layer of DHS of the data relay satellite system .....</b>     | <b>89</b> |
| Figure 85. Verification of RF links of DHS of the data relay satellite system .....                   | 90        |

## Abbreviation

|        |   |
|--------|---|
| ADCS   | Attitude Determination & Control Subsystem        |
| ANT    | Antenna   |
| AOBC   | ADCS Onboard Computer                             |
| AOS    | Atomic Oxygen Sample                              |
| ATOCSC | Antimony Tin Oxide Coating Solar Cell             |
| AVI    | Assembly, Integration and Verification            |
| C&DH   | Command & Data Handling                           |
| CCSDS  | The Consultative Committee for Space Data Systems |

|       |  |
|-------|--|
| CMD   | Command  |
| COM   | Communication  |
| DAS   | Data Acquisition System                              |
| DHS   | Data Handling System                                 |
| DoD   | Department of Defense                                |
| DoDAF | Department of Defense Architecture Framework         |
| EGSE  | Electrical Ground Support Equipment                  |
| EM    | Engineering Model                                    |
| EPS   | Electrical Power Supply                              |
| FFBD  | Function Flow Block Diagram                          |
| GMFE  | Generic Modular Front-end Equipment                  |
| GS    | Ground Station                                       |
| H/W   | Hardware   |
| I/O   | Input/ Output  |
| LNB   | Low-Noise Block                                      |
| LNBF  | Low-Noise Block Feed-horn                            |
| MCC   | Mission Control Center                               |
| MDG   | MicroDragon  |
| MDVE  | Model-based Development and Verification Environment |
| MIS   | Mission  |
| MOBC  | Main Onboard Computer                                |
| OV    | Operational View                                     |
| PC    | Personal Computer                                    |
| PCU   | Power Control Unit                                   |
| PDU   | Power Distribution Unit                              |
| RF    | Radio Frequency                                      |
| S/W   | Software   |
| SHU   | Science data Handling Unit                           |
| SMI   | Space-borne Multispectral Imager                     |
| STR   | Structure  |
| STRX  | S-band Transponder                                   |

|      |                                   |
|------|-----------------------------------|
| SV   | Systems View                      |
| SVF  | Software Verification Facility    |
| THR  | Thermal                           |
| TLM  | Telemetry                         |
| TLYF | Test Like You Fly                 |
| TOCT | Total Operations Chain Test       |
| TPI  | Triple Polarization Imager        |
| VNSC | Vietnam National Satellite Center |
| VSAT | Very Small Aperture Terminal      |
| WITL | Weeks In The Life                 |
| XTX  | X-band Transmitter                |

## Chapter 1 Introduction

### 1.1. Research Background

#### 1.1.1. Overview of satellite and ground segment

##### 1.1.1.1. Overview of satellite

A satellite is a moon, planet or machine that orbits a planet or star. For example, Earth is a satellite because it orbits the sun. Likewise, the moon is a satellite because it orbits Earth. In fact, the word "satellite" commonly refers to a machine that is launched into space and moves around Earth or another body in space <sup>[1]</sup>.

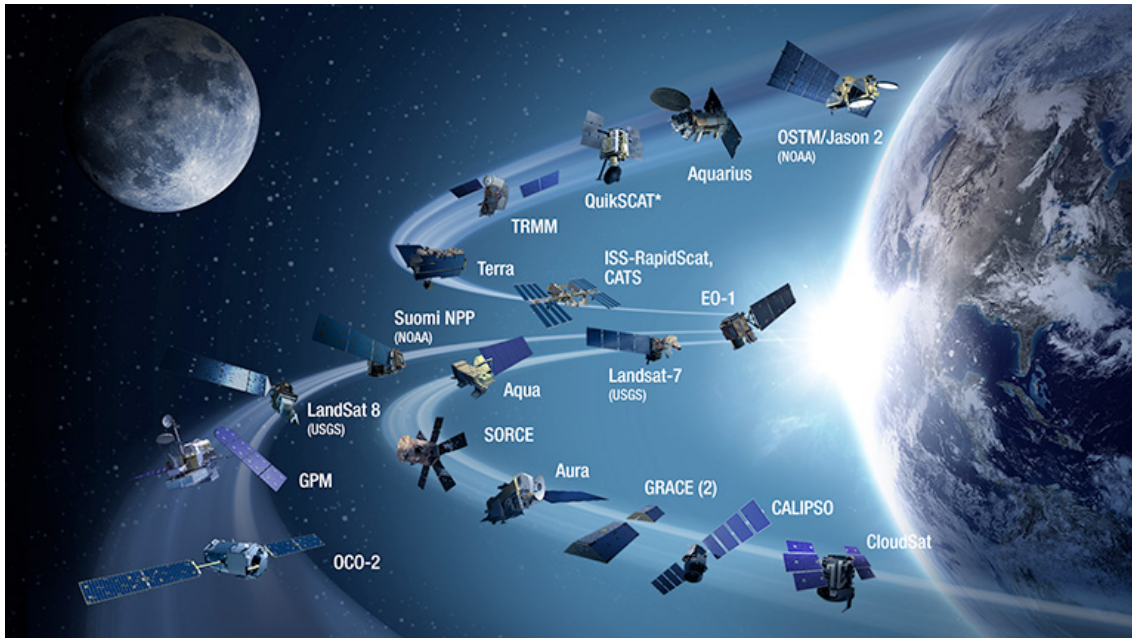


Figure 1. Satellites of NASA to study the oceans, land, and atmosphere [credits: NASA]

Satellites are developed in different shapes and sizes. Normally, one satellite consists of two parts including payload and bus. Payload instruments are used for implementing missions such as taking pictures, collecting weather information, etc. while bus components are assigned to some functions for controlling and maintaining operations of the satellite, such as, supplying electrical power, controlling temperature, protecting satellite from space environment, controlling and determining satellite attitude, receiving command from ground, sending back data to ground, controlling and managing operational activities.

As it is shown in Figure 2, one satellite commonly is composed of some subsystems which are assigned to specific functions, including Command & Data Handling system (C&DH), Attitude Determination and Control Subsystem (ADCS), Mission subsystem (MIS), Electrical Power Subsystem (EPS), Structure subsystem (STR), Thermal subsystem (THR), and Communication subsystem (COM).

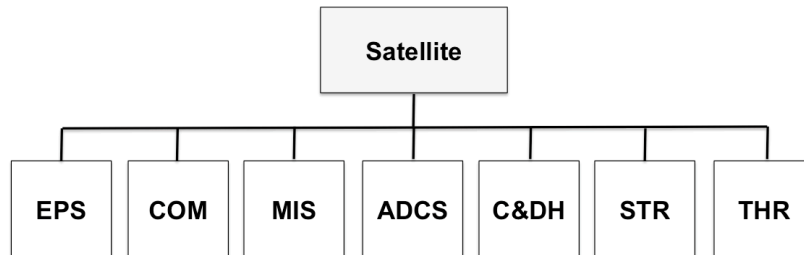


Figure 2. Common subsystems of a satellite

Satellites are launched into space by rockets. One satellite orbits the earth when the centrifugal force proportional to its speed is balanced by gravitational force caused by earth gravity. The satellite can travel around the earth in different altitude, different orbit shapes (circle or ellipse), and different orbital inclination compared with equator plane. Two common orbits are geostationary orbit and polar orbit. Geostationary satellites travel from west to east over the equator at altitude of 35,786km. They spin around the earth at the same spinning rate of the earth, therefore it seems that one geostationary satellite does not move if looking from ground. That is the reason geostationary satellites are usually designed to observe or cover one specific area on ground. Polar orbit is nearly perpendicular to equator plane. As the earth is self-spinning, polar-orbiting satellites can scan the whole globe when they travel from pole to pole.

To end life of one satellite, one way is that the satellite falls to the earth and be burnt by friction caused by the earth atmosphere. The other way is that it transfers its orbit to one disposal orbit, for geostationary satellites, disposal orbit is a few hundred kilometers about their operational orbit.

Nowadays, satellite-based applications and services grow strongly because of advantages of satellite-based technologies compared with ground-based technologies such as coverage, real-time properties, speed particularly, ability of remote sensing, communication, weather forecast, to name a few. With space technology, human can observe larger areas at one time, in other words, the satellite can collect more data, more quickly than ground-based instruments. For communication, the satellite can receive

television and phone calls signals then distributes to different locations on the earth. As it is indicated in Figure 3, number of satellites launched in 2014 increased 4 times from 2005 to 2014, especially it was 12 times in case of commercial satellites.

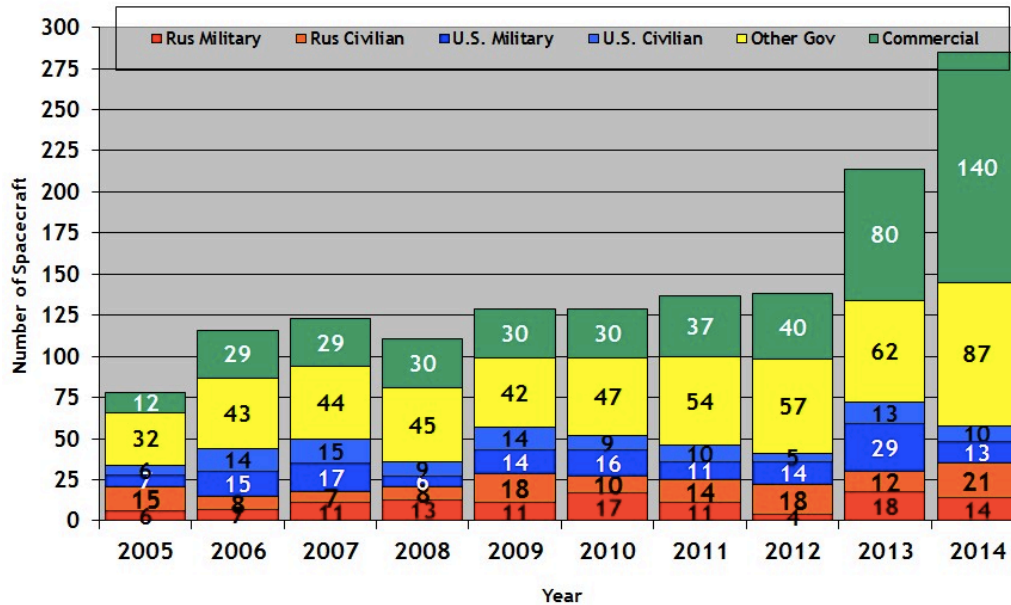


Figure 3. Number of Spacecraft launched from 2005 to 2014 <sup>[2]</sup>

#### 1.1.1.2. Overview of ground segment

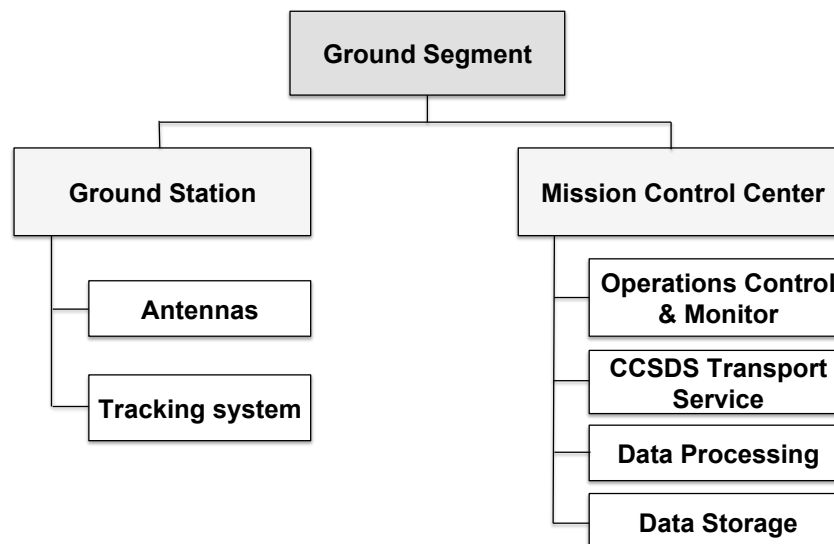


Figure 4. An example of ground segment

In order to operate the satellite as well as collecting data, it is necessary to have ground segment on ground. One ground segment usually includes a mission control center and one or more ground stations (including antennas and antenna control station), as it is illustrated in Figure 4.

A mission control center is a facility that manages space flights, usually from the point of launch until landing or the end of the mission. A staff of flight controllers and other support personnel monitor all aspects of the mission using telemetry, and sending commands to the vehicle using ground stations. Personnel supporting the mission from a mission control center can include members of C&DH, ADCS, THR, MIS, COM, and EPS subsystem. One example of the mission control center is the mission control center of international space station (ISS), shown in Figure 5.

A ground station (GS) includes antennas, which are used for transmitting data between the ground segment and the satellite, and a system for tracking the satellite and controlling the antennas. The antennas can be categorized into different groups depending on frequency of electromagnetic wave which carry the data, for example, X-band antenna, S-band antenna, C-band antenna, Ku-band antenna, to name a few.



Figure 5. International Space Station control room in Russia

Ground station antennas have a parabolic shape to increase possibilities of receiving electromagnetic signals from satellites. These signals can carry data or can be used for broadcasting. One parabolic antenna is designed to receive and transmit signals in specific frequency ranges, for example S-band antenna is used for S-band signals. Figure 6 illustrates one ground station of Galileo satellites located in Redu, Belgium.



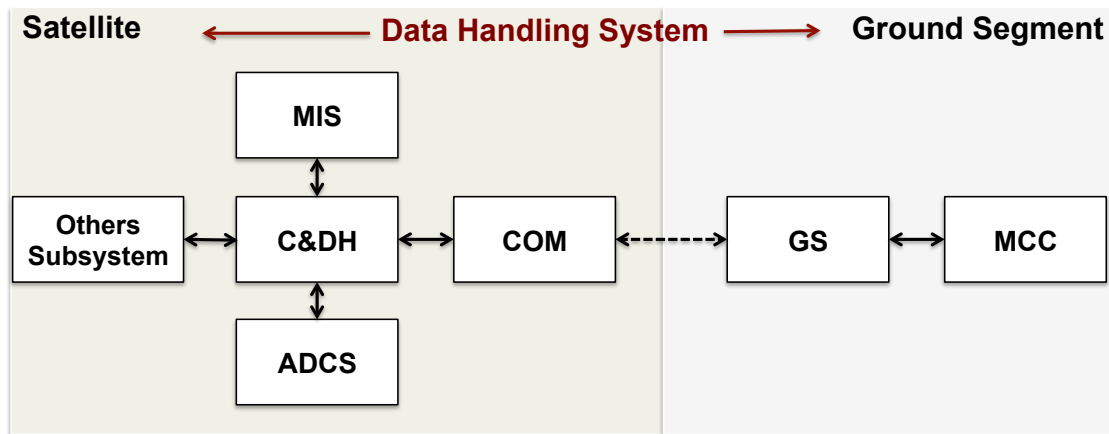
Figure 6. Ground station of Galileo satellites in Redu, Belgium

### ***1.1.2. Introduction to satellite Data Handling System***

#### ***1.1.2.1. Introduction to data handling system***

A satellite development project usually requires a great budget and other resources including human resources and facilities. However, many countries still want to spend a lot of money and effort on developing and launching the satellite because sometimes, it is necessary to collect data and information in real-time with bigger amount, and with as big coverage as possible, for example collecting weather information for agriculture development, disaster monitoring, communications, navigations, etc. As mentioned in the previous section, to acquire the data and information from satellites, it needs the ground segment. The data exchange system between flight segment and ground segment is defined in this thesis as Data Handling System (DHS). Figure 7 illustrates one example of data handling system.

In Figure 7, data handling system is responsible for collecting data in the satellite, followed by downlink these data to ground station, as well as generating and uplink command from ground station to operate the satellite. In general, data handling system is in charge of all the processes/ tasks related to data in the satellite and the ground segment.



**Figure 7. An example of Data Handling System between satellite and ground segment**

Data handling system operates based on communication subsystem in the satellite and in the ground segment. The communication subsystem is responsible for transmitting data between two segments, yet, it does not care about the content of data instead it only takes care of success of transmission. While DHS takes responsibility for that whether data received at one end, is exactly the same data sent from the other end. In other words, DHS takes care of all the data handling processes including data collection, data packet, data encryption, data processing, data decryption, data distribution, and data transmission. If one of these processes fails, DHS will fail.

There are three types of data going through DHS between satellite and ground segment including:

- Telemetry (housekeeping data) or health status information of the satellite: is defined as state information of satellite components. For example, ON/OFF status of a camera, temperature of onboard computer, current/ voltage of battery, etc.
- Command: are requests from ground station to take actions or change status of satellite components. For example, turn ON/OFF cameras, switch modes from coarse earth acquisition mode to fine earth acquisition mode, capture images, start data downlink, etc. Command is categorized into some groups, depending on execution timing and command structure, as shown in Table 1. Combining them together, four types of command can be created, including single real-time command, block real-time command, single timeline command, and block timeline command.

A single command executes single tasks, for example, turning on a camera. A

block command is a combination of single commands, performing a task, which requires multiple steps. For example, Figure 8 illustrates a block command which is sent from the ground station to the satellite to capture the pictures, is composed of 3 single commands including turn on camera → take one picture → turn off camera.

Table 1. List of command types

| <div style="text-align: center;"> <div>Structure</div> <div>Execution timing</div> </div> | Single                   | Block                   |
|---|--------------------------|-------------------------|
|   | Single real-time command | Block real-time command |
| Timeline  | Single timeline command  | Block timeline command  |

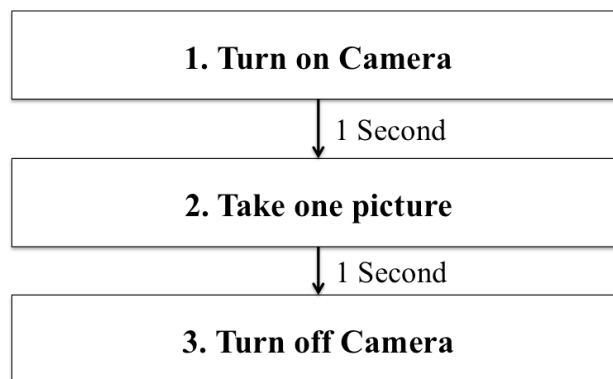


Figure 8. An example of a taking-picture block command

A real-time command is executed as soon as the satellite receives it from the ground station. As it illustrated in Figure 9, a timeline command is sent from ground station with time-to-execute information. The timeline command is normally used for the case when the ground station cannot see and communicate with the satellite. For example, the satellite has to take a picture of an area, however, at the time, the ground station cannot communicate with the satellite to send the command, thus, a timeline command is necessary in this situation.

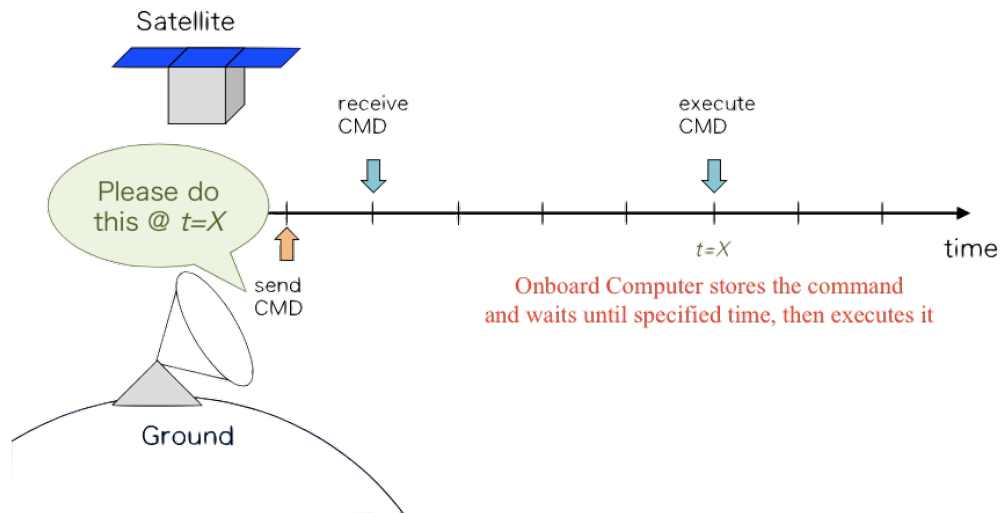


Figure 9. Execution of a timeline command

- Mission data: are data acquired from mission payload instruments of the satellite, for example images of ocean, atmosphere temperature information, atmosphere humidity information, etc.

To name one example, one satellite operational activity is described as follows. The ground station sends a block timeline command via uplink to tell the satellite to take pictures of a specific area on ground (the time when the satellite travels above the area to take the pictures is identified based on orbit calculation). When the satellite is above the area (at specified time), the satellite attitude will be determined and controlled appropriately, and the cameras will be turned on and take pictures of the area, then image data will be sent to and stored in the onboard computer. During the time that the satellite can be seen from the ground station, the satellite will send both telemetry and mission data via downlink to the ground station. In the ground station, the data are processed and sent to end-users.

#### 1.1.2.2. Data handling system lifecycle

Figure 10 illustrates the lifecycle of satellite DHS, which is divided into two phases: development phase and operations phase. The development phase starts from system requirements → system design → manufacture → verification and validation. In DHS development phase, the satellite or/and ground segment, are under development. As both segments are usually developed independently, it is difficult to confirm mutual understanding between those two segments. However, it is an essential step to verify whether the satellite in orbit has ability to talk with the ground segment on ground.

Therefore, verification of system interfaces is very important, particularly for development of data handling system. If upper-layer interfaces of data handling system can be determined and verified in low-level system development phase, it can reduce development risks by reducing unexpected iteration testing processes.

The context and the way, which data handling system operates in operations phase affects how this system is developed in development phase.

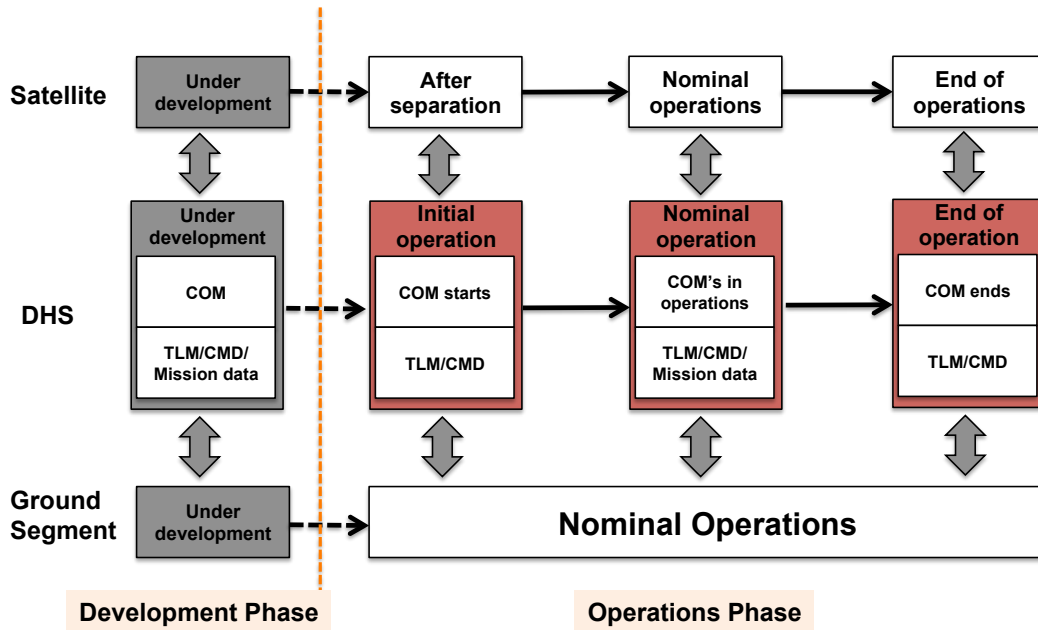


Figure 10. Lifecycle of satellite data handling system

As one ground segment is not designed for only one satellite, thus, its lifecycle is normally longer than that of the satellite. Therefore, the lifecycle of DHS is associated with the lifecycle of the satellite. The operations phase of DHS starts as soon as satellite communication starts, it is the moment right after separation of the satellite from the rocket, and ends when the satellite finishes its life.

### 1.1.3. Introduction to MicroDragon satellite

MicroDragon is a micro-class satellite of which weight and size are 50 kg, and 50cm x 50cm x 50cm, respectively. Figure 11 is a 3D-image of MicroDragon satellite. The satellite has four cameras categorized into two types: Space-borne Multispectral Imager (SMI), and Triple Polarization Imager (TPI). In addition, there are two secondary payloads developed at Kyushu Institute of Technology named Antimony Tin Oxide Coating Solar Cell (ATOCSC), and Atomic Oxygen Sample (AOS). As scheduled, the satellite is to be launched in 2018 by EPSILON rocket of Japan

Aerospace Exploration Agency.

The mission objective of MicroDragon satellite is to observe color of water in Vietnamese ocean areas in order to assess coastal water quality and locate aquatic creatures. The system will provide collected data for researchers and scientists in fishery field and oceanography for analysis and after that they will distribute necessary information to fishermen and environmental managers.

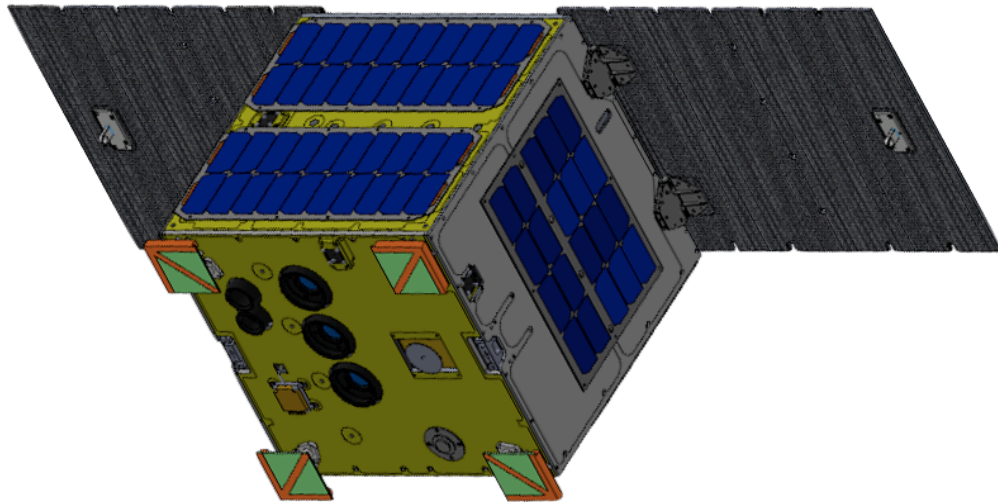


Figure 11. 3D-Image of MicroDragon satellite

Viet Nam has a coastline of 3,260 km that crosses 13 latitudes, from 8°23'N to 21°39'N. There are four main fishing areas: Gulf of Tonkin, shared with China; Central Vietnam; Southeastern Vietnam; and Southwestern Vietnam (part of Gulf of Thailand), shared with Cambodia and Thailand. Apart from these geographical zones, the fishing areas can be divided in inshore-coastal fishery and offshore fishery. Inshore water is considered as the water with the depth of less than 30m in the Tonkin Gulf and the South, and of less than 50m in the center of Vietnam. Vietnam has huge potential of coastal aquaculture with shrimp culture being dominant. Brackish-water shrimp is the main species raised along the coast. The enormous increase of the brackish-water aquaculture has some negative impacts as the silting of the inland area as the aquaculture areas are up to 10 km inland and the reduction of the mangrove area.

Vietnamese government has put forward the following policies: Using the state capital to invest in scientific research, building centers for aquatic breeding, building human capacity, and establishing stations for environment monitoring and forecast, fishery extension activities, food production and medicine for aquatic animals. The fisheries and aquaculture sectors are significant contributors to Vietnamese economy.

Seafood is always in top 10 of main exporting goods of Vietnam. There are about 7% households in the fishery. Vietnamese fisheries have been growing considerably and promoted by the government, aiming hunger elimination, poverty reduction, and increase of gross domestic product. However, aquatic natural resources are dwindling because of increasing exploitation. Therefore, the development of aquaculture is essential for sustainable economy. In order to develop aquaculture, it is necessary to know water quality, which is suitable for each aquatic species. However, monitoring of coastal water quality using only water samples taken in vast coastal area takes time and cost, besides it is not sufficient because of lack of necessary temporal and spatial information. Therefore, satellite technology with remote sensing technique is used for solving those problems.

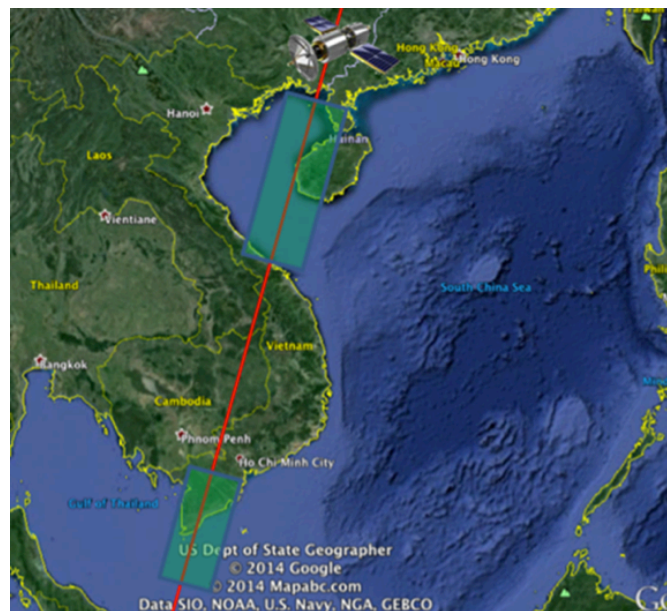


Figure 12. Nadir Scan Mode of MicroDragon satellite

MicroDragon satellite will take the pictures of Vietnamese ocean areas in several modes including:

- **Nadir scan mode:** As it is illustrated in Figure 12, the satellite will take images at nadir during the period of time flying over the sea of Vietnam. The nadir is the direction pointing directly below a particular location. It is orthogonal to a horizontal flat surface there. The nadir at a given point is the local vertical direction pointing in the direction of the force of gravity at that location.
- **Tilt scan mode:** The satellite will take images in off-nadir direction during the time flying over Vietnamese sea area, as it is shown in Figure 13.

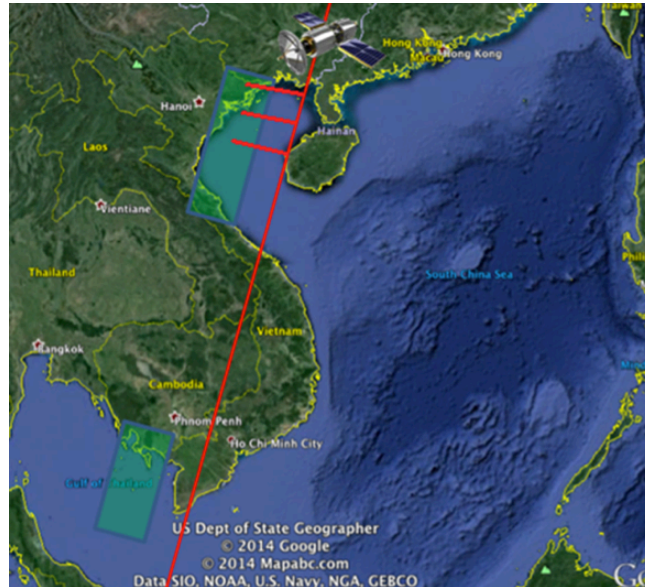


Figure 13. Tilt Scan Mode of MicroDragon satellite

- **Target scan mode:** This mode is illustrated in Figure 14. The satellite will take images of desired targets when it flies over Vietnamese sea area.

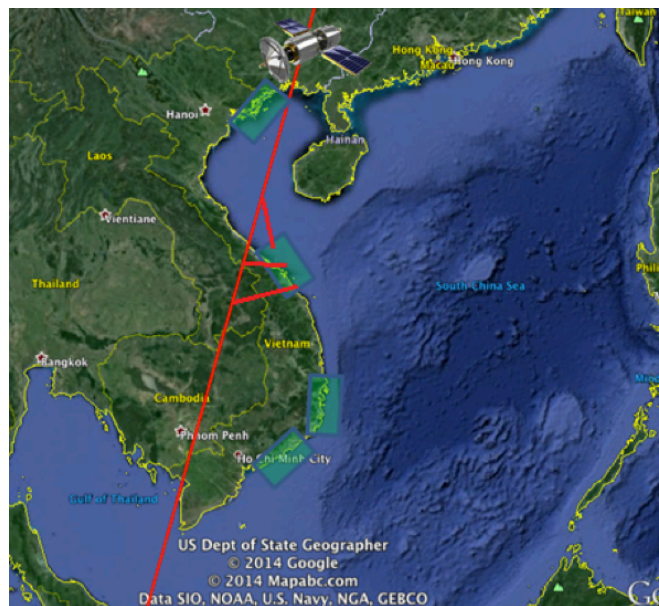
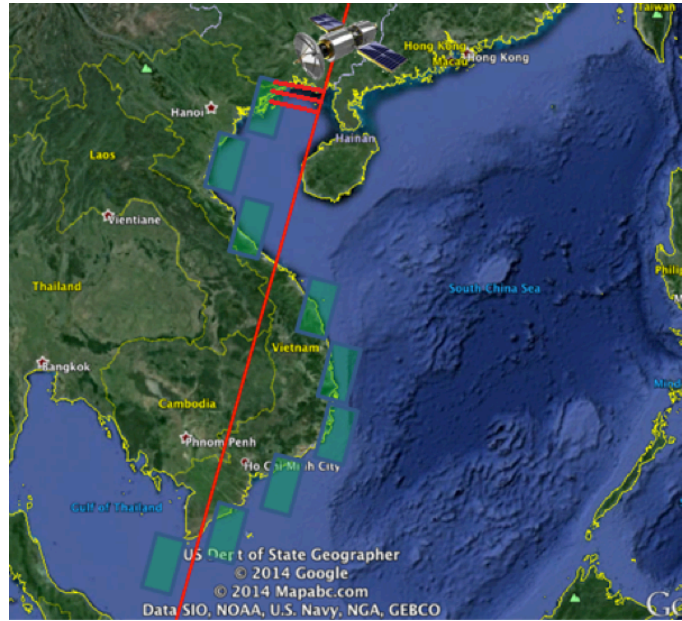


Figure 14. Target Scan Mode of MicroDragon satellite

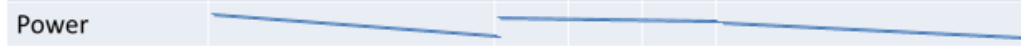
- **Coastal scan mode:** The satellite will take images along Vietnamese coastal line, as it is shown in Figure 15.



**Figure 15. Coastal Scan Mode of MicroDragon satellite**

The concept of operations of MicroDragon satellite is as follows:

- For primary mission payloads: Operations commands are sent to the satellite to operate the cameras twice a day during daytime. Data downlink can be done automatically and manually.
- For secondary mission payloads: Figure 16 describes the conceptual scenario of secondary mission of MicroDragon. The measuring circuits are turned on for 5 minutes once a month. In each time, data will be collected and downlinked.

|                   | Day 1  | Day 2 | ... | Day 30 | Day 31           |
|-------------------|--|-------|-----|--------|------------------|
| Measuring circuit | ON in 5 minutes  | OFF   | OFF | OFF    | ON in 5 minutes  |
| Power             |  |       |     |        |                  |
| Communication     | Downloading data   |       |     |        | Downloading data |

**Figure 16. Conceptual scenario of secondary mission of MicroDragon satellite**

#### **1.1.4. Introduction to TableSat**

TableSat is referred to testing facilities developed for verification of functional and operational design of a satellite. At Tokyo University, the TableSat facilities including the ground segment and the satellite are arranged in two different rooms, as it is illustrated in Figure 17.

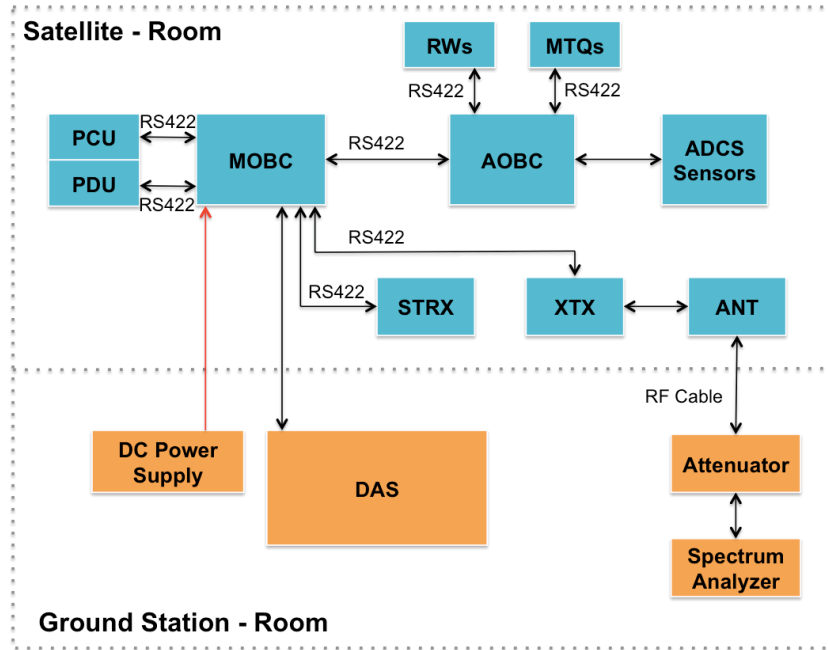


Figure 17. TableSat layout at Hongo campus, Tokyo University

In the satellite room, engineering model (EM) of satellite components are set up on a table, such as X-band Transmitter (XTX), S-band Transponder (STRX), Antenna (ANT), Main Onboard Computer (MOBC), ADCS Onboard Computer (AOBC), Power Control Unit (PCU), Power Distribution Unit (PDU), ADCS sensors, Reaction Wheels (RWs), Magnetic Torquers (MTQs), to name a few. These components may be connected or disconnected depending on purposes of specific functional/ operation tests. Figure 18 is the image of TableSat satellite room at Tokyo University.

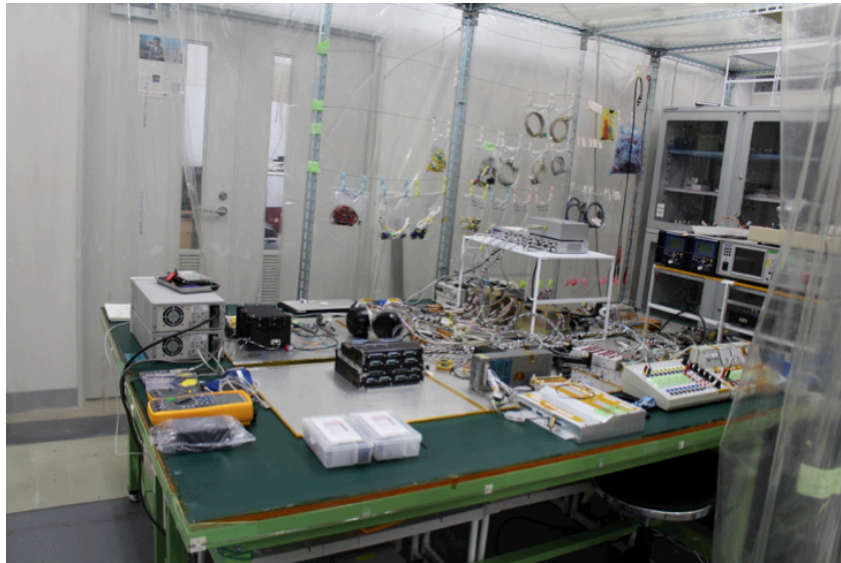


Figure 18. TableSat satellite room at Tokyo University

In ground station room, there are a Data Acquisition System (DAS), a DC power

supply, and some devices, which are used to collect and display data. DAS includes an operations computer of which functions are to send command and to display telemetry, and some telemetry analysis computers. Figure 19 is an image of data acquisition system.



Figure 19. TableSat data acquisition system at Tokyo University

## 1.2. Research Objective

For a development of data handling system, especially a new system, it is important to confirm understandings among subsystems in a system but for a new system, it becomes very difficult because it may miss to identify some upper-layers interfaces<sup>[28]</sup>. Missing of identification of upper-layers interfaces leads to increase of iteration testing processes and development risks.

The objective of the research is to reduce iteration testing processes and development risks by introducing an approach to design of satellite data handling system for architectural-layer-driven verification. The approach clarifies a systematical way of designing process of data handling system, which supports verification of upper-layers data interfaces at low-level testing. The approach helps to reduce development risk and unexpected iteration testing processes, as before these upper-layers interfaces are normally tested based on experience without explicit systematic way.

As it is seen in development V-model of DHS in Figure 20, the development process of data handling system starts from concepts of operations and analysis of system requirements, goes to design of the system, followed by subsystems/

components design, and manufacture of hardware. Verification of DHS design can be done through some testing levels including low-level testing and system-level testing. One critical issue of testing DHS is confirmation of upper-layers data interfaces, which cannot easily be done in low-level, is solved in this research. Verification of upper-layers interfaces at low-level testing helps to shorten unexpected iteration testing processes and it also help to confirm understandings among subsystems in early phases which helps to reduce development risks.

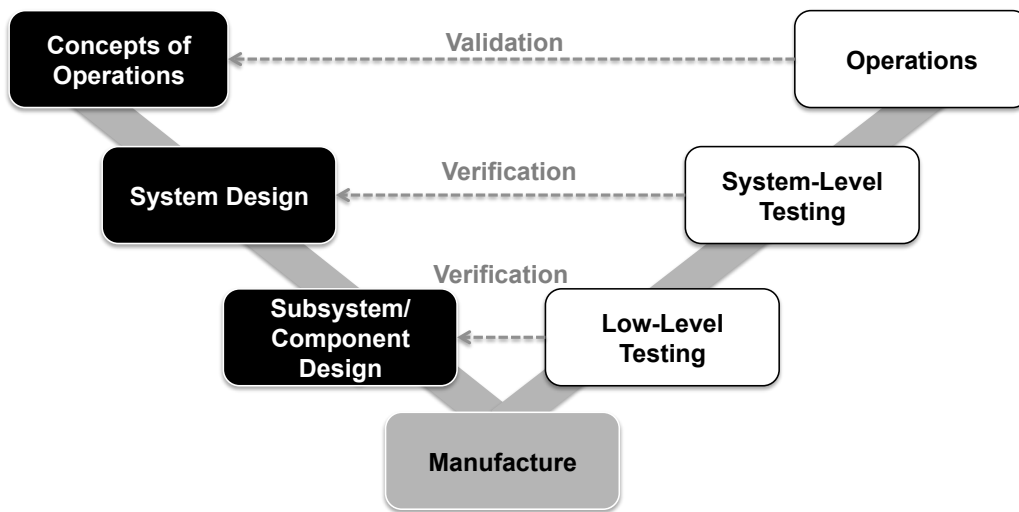


Figure 20. Development V-Model of Data Handling System

### 1.3. Scope of Research

The research focuses on introducing the approach to satellite data handling system design for architectural-layer-driven verification to reduce iteration testing processes and development risks of data handling system.

The approach clarifies a systematical process of designing and verifying data handling system. However, the approach focuses on new designing process rather than new verification configuration of data handling system, as these configurations have already been stable and saturated in space industry.

For evaluation, the approach is to be applied to design and verify data handling system of a Micro-satellite, and a data relay satellite system. Besides, the evaluation is also done, by interviewing five satellite specialists from different space backgrounds.

### 1.4. Outline of dissertation

This master's dissertation includes 5 chapters as follows:

In chapter 1, research background is introduced. This chapter starts with overview of satellite and ground segment including definition of satellite, how satellite is launched into space, how it survives and orbits the earth, how it ends its life, and components of ground segment as well as its operation. In latter parts, satellite data handling and its lifecycle are described in detail, followed by introduction to MicroDragon satellite, and TableSat.

In chapter 2, literature review is given. This chapter is focusing on existing approaches to testing satellite in general and testing data handling system in particular, together with the advantage and disadvantage of those approaches. The final section in this chapter shows discussion and conclusion.

In chapter 3, an approach to data handling system design for architectural-layer-driven verification is introduced. This chapter starts with proposal of the approach and its originality. Detailed approach is the main content of this chapter.

In chapter 4, evaluation of the approach is described in more details. The first way of evaluation is to apply the approach to design and verify data handling system of MicroDragon satellite, and that of a data relay satellite system. The second one is to interview five satellite specialists from different space backgrounds.

In final chapter, conclusion and discussion on the research and future work are given.

## Chapter 2 Literature Review

### 2.1. Survey of existing approaches to satellite development

#### 2.1.1. Model-based development and verification (MDV)

##### 2.1.1.1. Overview of Model-based Development and Verification Environment (MDVE)

MDVE <sup>[7]</sup> has been developed by Astrium company since 2001 with the goal to reduce development time & cost, and to improve systems engineering process in satellite development. The process of developing satellite can be improved based on simulation-based software and hardware environment, which is able to support different development phases from B to E, as it is shown in Figure 21.

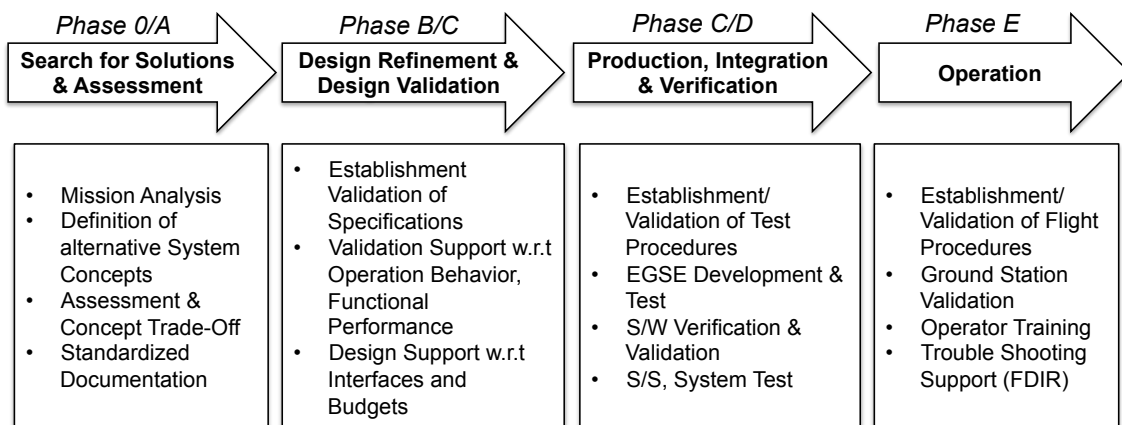


Figure 21. Satellite development and operation phases defined in MDV <sup>[7]</sup>

To name a few, some functions of MDVE are listed as follows

- Early validation of operational design, functional design, and their interfaces. Early validation of ground operations, and electrical ground system equipment (EGSE). These functions allow to early validation of satellite data handling system.
- Evaluation of onboard software
- Support designing process of satellites

Some important elements included in MDVE are shown in Figure 22.

- A real-time system simulator, which contains models of the satellite, and its context environment, such as models of satellite subsystems, and models of space

environment.

- An Electrical Ground Support Equipment (EGSE) system controls the satellite, and the real-time system simulator.
- Generic Modular Front-end Equipment (GMFE) consists of a set of input/output (I/O) cards to connect real hardware of satellite components to the real-time satellite simulator.
- Software Verification Facility (SVF) for evaluation of onboard software.

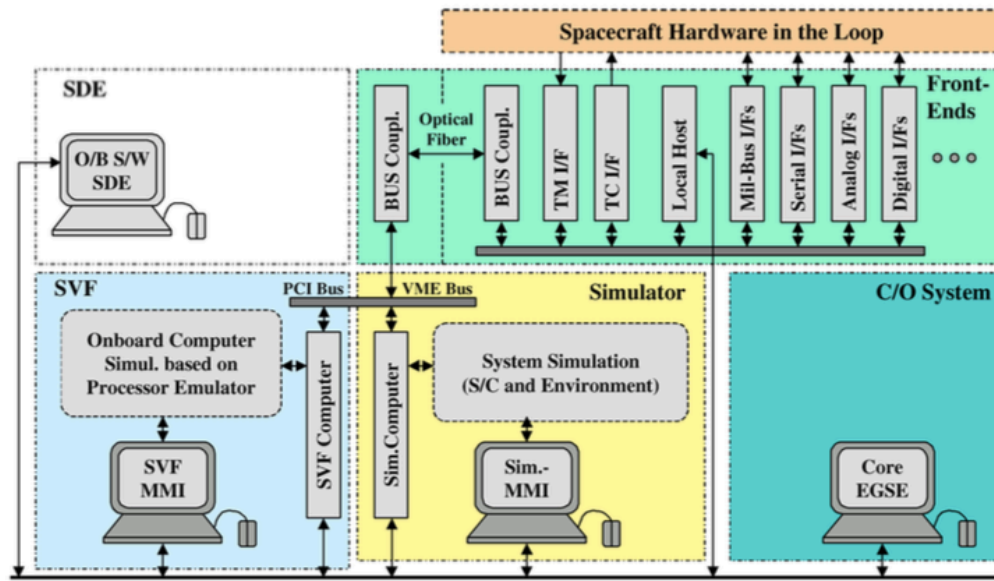
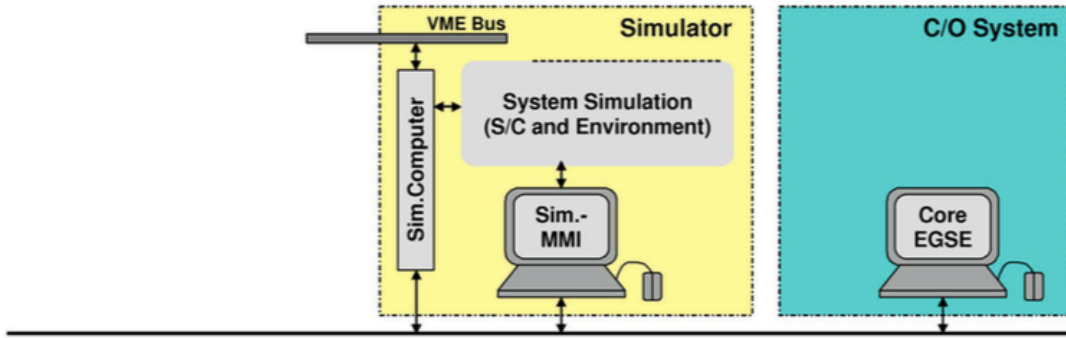


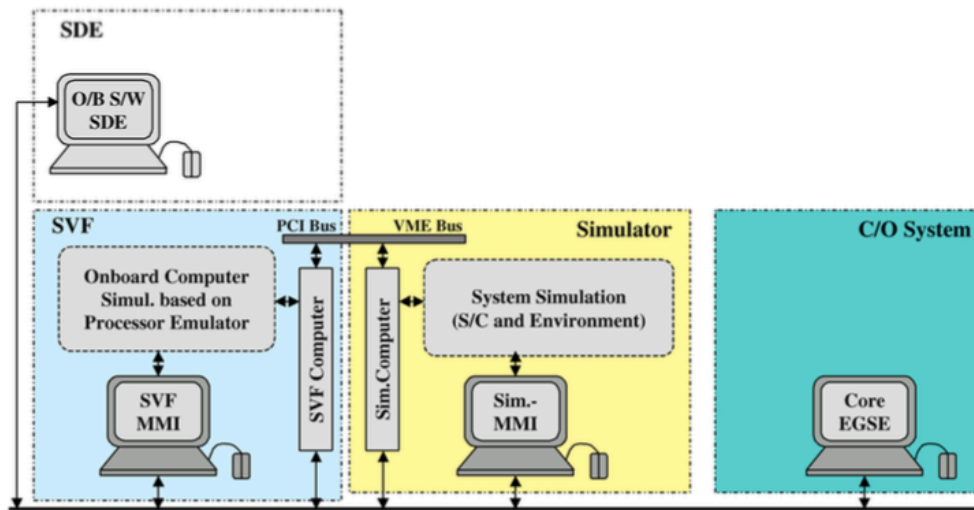
Figure 22. MDVE configuration <sup>[7]</sup>

#### 2.1.1.2. Systems engineering process and application of MDVE

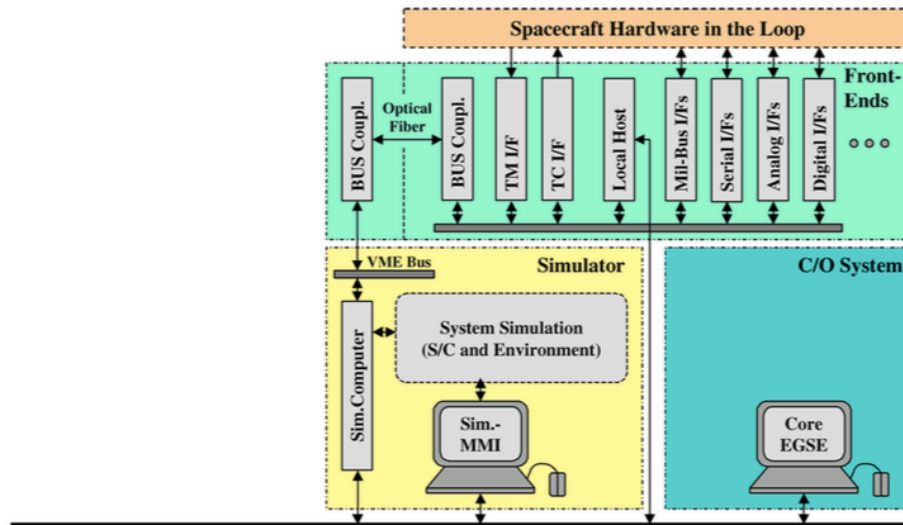
a) **Phase B (Design refinement and checkout system setup):** Figure 23 shows the MDVE configuration in phase B, which includes the real-time satellite simulator, and EGSE. EGSE is used for controlling and operating the satellite simulator by sending command, and receiving relevant telemetry.

Figure 23. MDVE configuration in phase B <sup>[7]</sup>

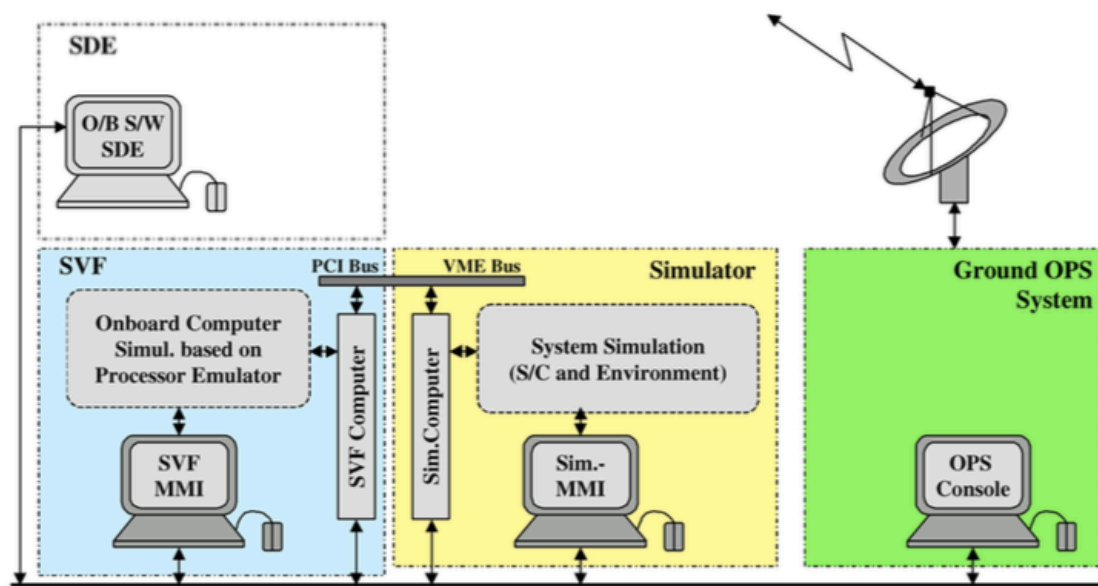
b) **Phase C (Production):** Figure 24 illustrates the onboard software verification facility (SVF) configuration of MDVE in phase C, including a real-time satellite simulator, an real-onboard-software-embedded onboard computer simulator, and a man machine interface (MMI) to send commands, analyze telemetry, debug onboard software, inject failures test for onboard software, to name a few.

Figure 24. MDVE configuration in phase C <sup>[7]</sup>

c) **Phase D (Spacecraft Assembly, Integration & Verification):** Figure 25 shows the MDVE configuration in phase D, which is composed of EGSE, satellite simulator, and satellite hardware-in-the-loop (HIL). These elements are connected via front-ends ground equipment.

Figure 25. MDVE configuration for phase D <sup>[7]</sup>

d) **Phase E (Operations support):** Figure 26 illustrates configuration of MDVE in phase E. In this phase, MDVE can support satellite operators training, identification of operations failures of satellite, evaluation of onboard software before uplink to the satellite in orbit.

Figure 26. MDVE for phase E <sup>[7]</sup>

### 2.1.2. “Test like you fly” approach

#### 2.1.2.1. Introduction to “Test like you fly approach”

##### a) Overview

“Test like you fly” (TLYF) <sup>[25]</sup> is an approach that provides a unique assessment

process that focuses on determining the “mission-related” or “like you fly” risks associated with potential flaws in our space systems.

TLYF goes beyond test. This approach contains strategy of acquisition, development of products, definition of requirements, systems engineering, fault analysis, and risk management. As systems engineering methodology often focuses on verifying requirements or confirming that a system meets its requirements, this methodology fails to emphasize test capability, which is necessary to find flaws in an actual system. While TLYF focuses on demonstrating that a mission can be implemented successfully. Procedures of TLYF tests are derived from concept of operations, concepts of mission operations, rather than system requirements. Many systems can successfully meet their stated system requirements but fail to implement the mission. Therefore, TLYF tests are introduced to confirm whether the space and ground systems accomplish the mission.

b) Detailed TLYF approach

TLYF can cover any of the following methodology, process, tool, and test.

- A systems engineering methodology which focuses on validation of mission performance ability of system, rather than verification of system requirements
- A process to determine testability of mission concepts, and to identify the risk of untestability: The process is illustrated in Figure 27. “Like you fly” testing is driven by concepts of operations, flight requirements/ constraints, flight rules/ procedures, and users needs. If the mission can be tested, by applying “test like you fly” approach, “like you fly” tests includes identification of test plans, test resources, test procedures, and test configurations. For untestability case, “Test like you fly” exceptions and risks should be accessed.

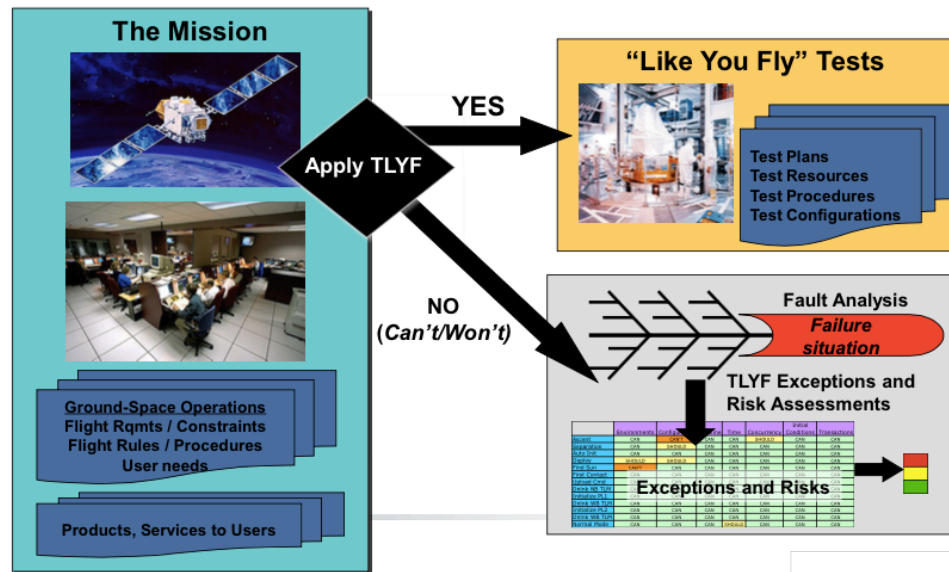
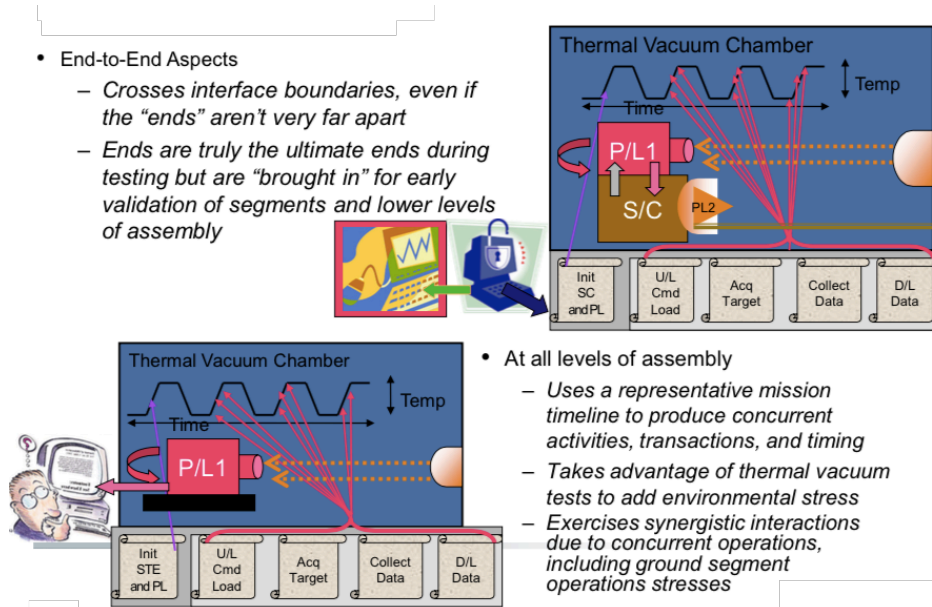
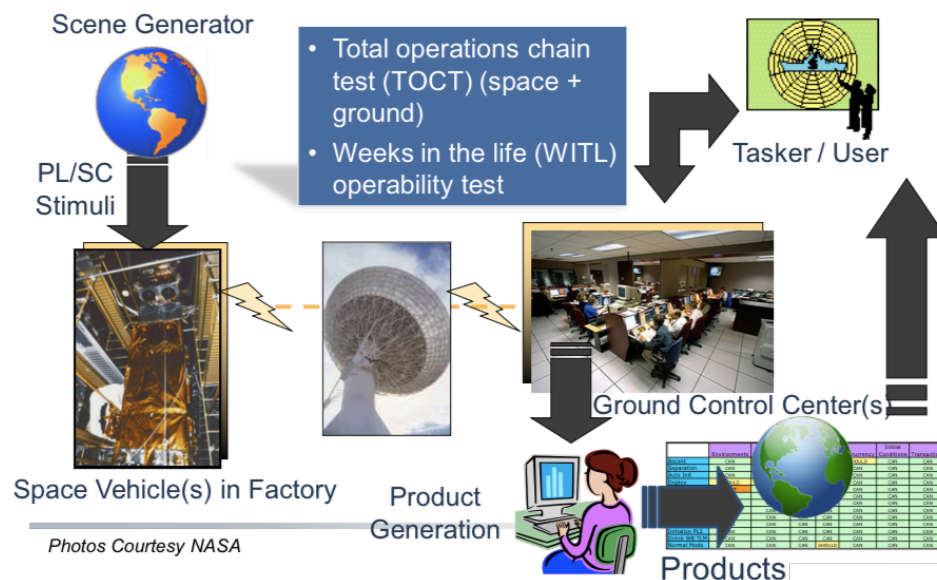


Figure 27. TLYF process to access mission concepts for testability and to access risk <sup>[25]</sup>

- A validation tool of mission performance: TLYF is the approach, focusing on validation of mission performance, instead of that of system requirements.
- A test technique for mission operability at all levels of assembly, as it is described in Figure 28: Operability testing is a good way to discover system flaws, such as data errors, timing issues of software/hardware, memory leaks, to name a few. A mission operability test is different from a single functional and performance test, it concentrates on testing mission performance under specific mission conditions, and timelines. Mission conditions include space environment conditions, such as thermal, radiation, vacuum, micro-gravity environment. Timelines indicate smooth transition from one activity to another, timing interactions between asynchronous activities, to name a few.

Figure 28. TLYF operability test technique <sup>[25]</sup>

- An operations readiness test, illustrated in Figure 29: The operations readiness test is defined as a total operations chain or end-to-end (space and ground) days-in-the-life/ weeks-in-the-life operability test. End-to-end tests can detect defects that cannot be found in any other tests, such as electrical, functional, and performance tests.

Figure 29. TLYF total operations chain test <sup>[25]</sup>

In total operations chain test, inputs come from payload and satellite, and outputs are mission products (images, data, to name a few), mission services (for example, communications, navigation).

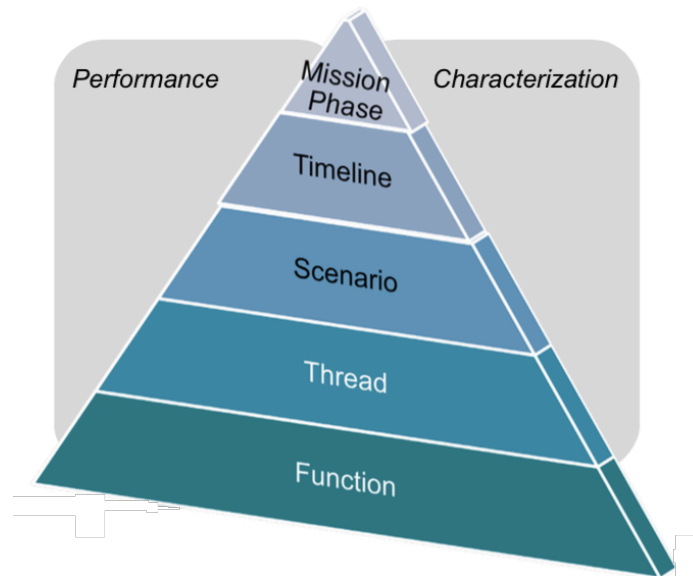


Figure 30. Key to injection of “like you fly” tests <sup>[25]</sup>

Key to injection of “like you fly” tests is described in Figure 30 as follows:

- *Function*: is an action or activity related to mission, such as sensing rocket launch, routing communication signals between users, commanding a satellite, etc.
- *Thread*: is either a stimulus or behavior that results from a sequence of system level inputs, and outputs.
- *Scenario*: is a prediction of system uses from a user perspective.
- *Mission timeline*: includes activity order, and timing.
- *Mission phase*: is distinguishable by a discrete change in a key characteristic.

#### 2.1.2.2. Distinction between TLYF and other tests

Table 2 describes the difference between TLYF and other tests.

Table 2. Distinction between TLYF and other tests

| Test  | Features   |
|---|--|
| <i>Typical functional, performance, and compatibility testing</i> | This is requirements centric test, focusing on verification of requirements.   |
| <i>Environmental testing</i>                                      | This includes qualification test of dedicated engineering-model hardware to verify the design, and acceptance test of flight |

|  |   |
|--|---|
|  | hardware to verify workmanship defects. However, demonstrating that hardware survives space environment, is different from showing that the integrated systems/processes work properly.   |
| <i>Requirement verification testing</i>  | Top-level requirements are usually oriented to mission specific performance characteristics, such as resolution, antenna gain, images per pass, bit error rate, data latency, etc. The requirements are usually derived from end-users needs. The system can meet system requirements but it does not guarantee that the system can meet requirements of mission operations where time, timing, order, transition, and environmental interactions, may affect the product or service. |
| All the 3 testing types, mentioned above, are necessary and valuable to find out specific flaws and defects, but not sufficient for validation of mission performance. |   |
| <i>Test like you fly</i>   | This test focuses on mission operations. The lessons show that it is necessary to include “mission operations centric” tests, which demonstrates the capability of mission performance of an integrated system in pre-launch phase <sup>[25]</sup> . TLYF test may (not must) cover any of 3 fore-mentioned testing types.  |

### ***2.1.3. Orbital Sciences spacecraft integration and test suite based on “test as you fly” approach***

With the evolution of geo communications product, Orbital Sciences has had an opportunity to develop, evolve and enhance its spacecraft integration and test suite that used operator-driven manual commanding to repeatable, automated test sequences executed with minimal operator intervention based on “test like you fly” approach. These enhanced procedures have been implemented in such a way that the same core test suite can be used across multiple spacecraft builds with a minimum of changes from one program to the next <sup>[3]</sup>. The goal is to solve three test problems of geostationary telecommunications satellite as follows:

- Validation of command through accessing telemetry
- Improved testing setup to simplify execution, repeatability and analysis of attitude control system operational tests

- Implementation of an automated method to simplify spacecraft unique heater and thermistor testing

The principle of the test is that verification of basic hardware functions is done, by accessing telemetry responses of relevant hardware commands. At system level, 50% of discrepancies are shared by 25% of operator error, and 24% of design error. A repeatable, automated test suite can significantly reduce the portion of discrepancies due to operator error <sup>[3]</sup>. Up to 2007, spacecraft integration and test suite in Orbital Sciences had successfully tested, launched, and verified in-orbit ten commercial geostationary telecommunications satellites such as Indostar-1, BSAT-2a, BSAT-2b, BSAT-2c, Star-2, to name a few <sup>[3]</sup>.

“Test as you fly” approach is performed via an operational ground control station. All test actions are initiated by flight commands, and responses are validated by flight telemetry. To provide repeatable testing, test scripts have been developed in Spacecraft Test Operations Language.

#### *2.1.3.1. Telemetry access for command validation*

One persistent problem of satellite command validation is location and frequency of the telemetry data <sup>[3]</sup>. Many telemetry points, which are used for diagnosing hardware or flight software errors, are not always necessary during normal operations. Therefore, they should be collected less frequently than critical telemetry points.

In any selected modes, all the required telemetry should be available in a format. However, during satellite testing, a specific telemetry required for command validation may be not available in active formats. If test operator does not know where to find the telemetry, telemetry access will become more difficult. Orbital Sciences has developed “Telemetry Point Retrieval Logic” algorithm to solve this problem.

The algorithm is described in Figure 31 as follows. The telemetry locations are identified by bit map variables attached to the telemetry point names. When a telemetry point is requested, the bit map will detect whether the point is available in either of two active formats. If so, returning the point to the operator. If not, scanning the bit map to find the format, which includes the point, and the system command format change on stream 2, followed by commanding the stream 2 back to original format after retrieving the requested point.

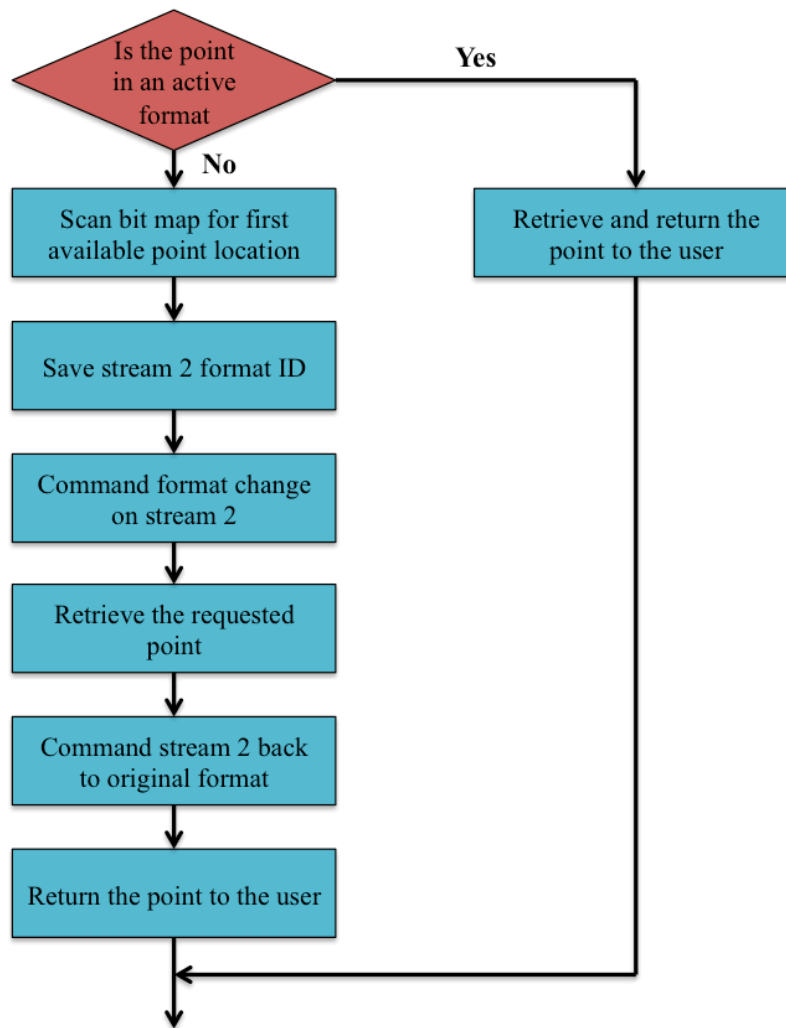


Figure 31. “Telemetry point retrieval logic” algorithm <sup>[3]</sup>

#### 2.1.3.2. Improved setup to simplify execution, repeatability and analysis of attitude control system operational tests

Because of complexity of Attitude Determination and Control System (ADCS), and its dynamic properties, ADCS testing results cannot be repeatable. Besides, some of results need to be post-processed to evaluate test success or test failure. To solve those issues, several changes have been applied to testing configuration of ADCS as follows:

- Changes of ADSC flight software: Introduction of commands, used for setting parameters in the test environment, supports a better control over ADCS onboard software.
- Improvement of repeatability of the tests by re-making the test scripts: Test scripts have been re-worked to perform additional operations. This reduces overall test duration by reducing necessary intervention of operators.

### 2.1.3.3. Implementation of an automated method to simplify spacecraft unique heater and thermistor testing from one spacecraft to the next

Thermal Control System (TCS) health check provides necessary validation tests of functionality of all the heaters and thermistors on the satellite. Typical TCS test sequence is in Figure 32. All the parameters required to test heaters are shown in Table 3.

**Table 3. Heater test data elements <sup>[3]</sup>**

| Item                         | Description  |
|------------------------------|--|
| Script number                | An unique identifier of heater circuit   |
| Circuit identifier           | Identifier of TMON circuit for the heater  |
| Description                  | Title used in test log   |
| Enable command               | Command to enable the heater circuit   |
| Disable command              | Command to disable the heater circuit  |
| Enable telemetry             | Telemetry to validate success of enable/ disable command                         |
| ON command                   | Command to turn on the heater  |
| OFF command                  | Command to turn off the heater   |
| ON telemetry                 | Telemetry to validate success of ON/ OFF command                                 |
| Control sensor               | Name of telemetry associated with the heater                                     |
| Number of control sensors    | Number of telemetry used to control the heater                                   |
| Additional sensors           | Name of additional telemetry associated with the heater                          |
| Bus current telemetry source | Identifying which current sensor is used for detecting the heater current change |
| Heater resistance            | Resistance of the heater   |
| Current resolution           | 1-bit measurement resolution   |
| Resistance tolerance         | Acceptable tolerance for resistance calculation                                  |
| Set point ON Delta           | TMON ON set point to trigger the heater ON                                       |
| Set point OFF Delta          | TMON OFF set point to trigger the heater OFF                                     |
| TMON control information     | Additional points required to manage TMON control processing                     |

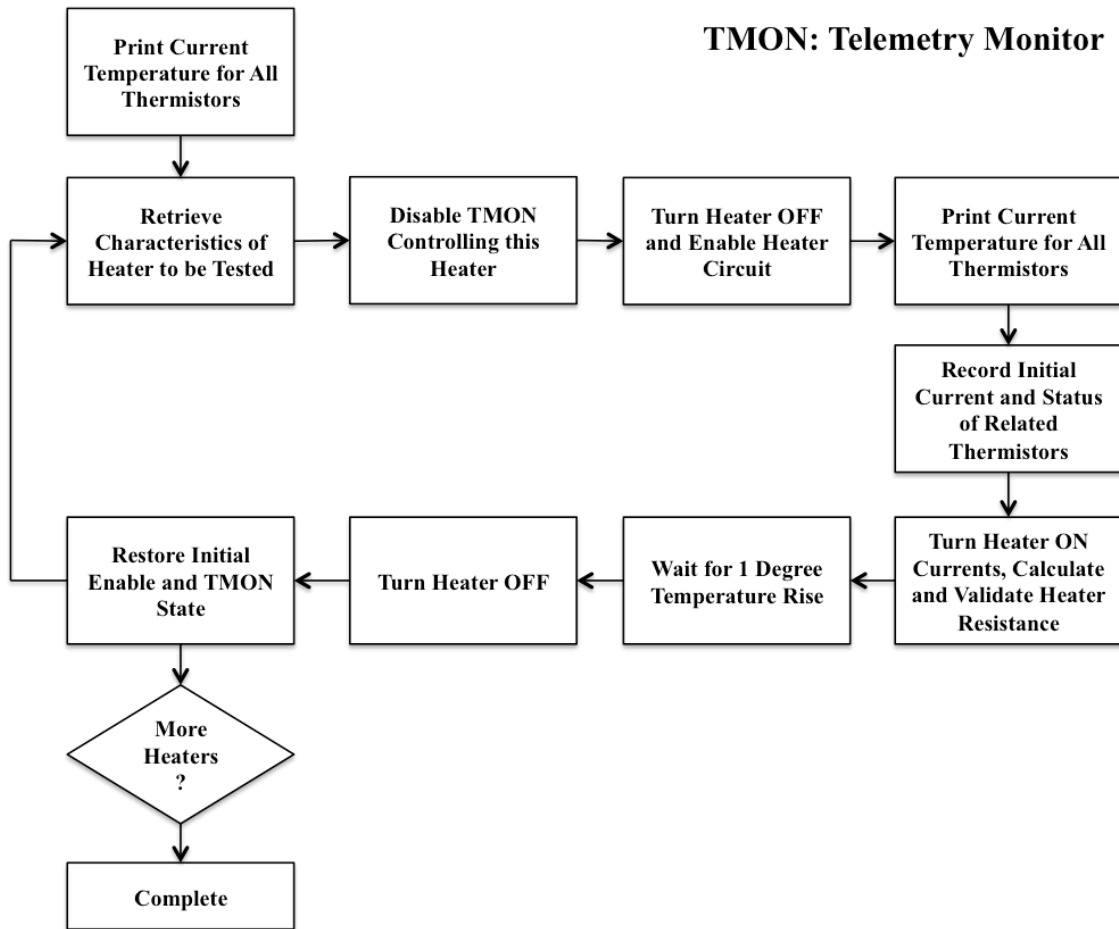


Figure 32. Thermal Control System Health Test Sequence <sup>[3]</sup>

#### 2.1.4. Best practices for verification, validation, and test in model-based design

The paper <sup>[22]</sup> focuses on using model-based design in verification, validation of on board software. As the size and complexity of the embedded software programs grow, using traditional development tools such as editors, compilers, and debuggers, is not suitable anymore, thus, is substituted by models. The models can be used

- As executable specifications
- To define system/subsystem requirements, and related interfaces
- To visualize virtual prototypes, and complete system
- To automatically generate codes of onboard software algorithms

##### 2.1.4.1. Developing model tests with the design

One advantage of model-based design is using models to perform verification, validation, and test. It means that test suites can be applied to the models, and the test suites can be used to test the onboard software implementation. The entire test suite can

be used repeatedly for onboard software testing. This process is called model-in-the-loop-simulation (MILS). After that, the same MILS are implemented again on the host development environment to make the software-in-the-loop-simulation (SILS). To test the compatibility of onboard software and hardware, the software can be embedded in targeted processor after compiling. The co-simulation step is called processor-in-the-loop-simulation (PILS) or hardware-in-the-loop-simulation (HILS). All these processes are illustrated in Figure 33.

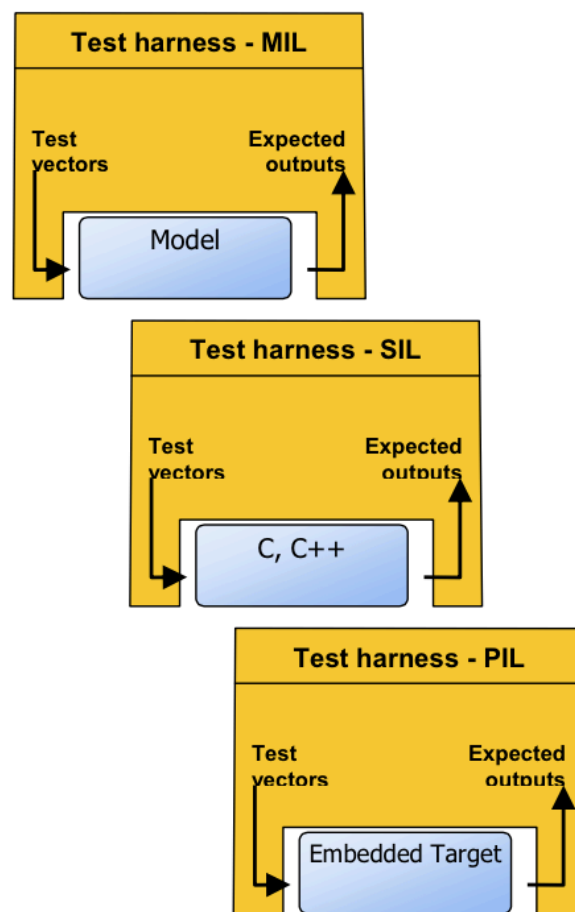


Figure 33. Test suites running against MIL, SIL, and PIL <sup>[22]</sup>

#### 2.1.4.2. Testing exhaustively in simulation

Most engineers believe more in results of hardware testing, however the hardware testing may be too costly since if some hardware elements do not work properly, the prototypes will be damaged or destroyed. While with testing in a model environment, many different test cases can be run faster, besides it also can help to reduce the number of critical tests, which must be done with hardware in real time.

#### *2.1.4.3. Running the same tests in simulation and in the lab*

To run the same tests in simulation and in the lab, it is necessary to have a facility, which supports hardware connectivity, measures physical quantities, and link to the modeling environment. The same testing contexts have to be used in order to analyze hardware and simulation test results.

Hardware-in-the-loop testing should be done before the availability of manufactured hardware. It is used to test production electronic control units (ECUs) and to test prototype control algorithms.

Integration testing: Model-based design can be used to make continuous testing and validation process. This process begins from testing of the lowest level hardware (component) to testing of the complete system. It helps to detect defects in the interface among components, subsystems in early phases.

## **2.2. Discussion and conclusion on the existing approaches**

All the reviewed approaches have shown their advantages of testing data handling system. MDVE focuses on verification of design while “test like you fly” focuses on verification of mission performance. However, for development of a new system, mutual understanding among subsystems/ systems is very critical, thus, it should be confirmed as early as possible. The more interfaces are confirmed in early phases, the more development risks and failures can be reduced.

All the fore-mentioned approaches miss to clarify a systematical process to design and verify a system in such a way that upper-layers interfaces of system of interest, data handling system in this research, can be verified at low-level testing.

MDVE <sup>[7]</sup> focuses on development and verification environment, testing facilities are built to support development of satellite system. MDVE supports development but it cannot show designing and verifying processes of a system.

“Test like you fly” approach <sup>[25]</sup> target at validation of mission performance in space environment, rather than at verification of system performance.

Orbital Sciences test suite <sup>[3]</sup> supports validation of command, attitude determination and control subsystem tests, and thermal control subsystem tests, rather than systematical designing and testing processes of a system.

Model-based design for verification, validation, and test approach <sup>[22]</sup> support evaluation of onboard software, focuses on testing processes during development phases.

This approach is very suitable for development of attitude determination and control subsystem (ADCS) where ADCS algorithms need to be tested before being applied to flight software. However, it is not suitable for development of data handling system, which consists of hardware, onboard software, and all data handling processes existing in the satellite and ground segment.

In conclusion, a systematical approach to design and verify satellite data handling system is necessary. In the next chapter, this approach is to be introduced. The designing and verifying processes of data handling system are explained in detail. The goal of the approach is to reduce development risks of increasing development time and cost by shortening unexpected iteration designing and testing processes. The iteration processes can be reduced, by verifying upper-layers interfaces at low-level testing.

---

## Chapter 3 Proposal: Satellite Data Handling System Design for Architectural-Layer-Driven Verification

### 3.1. Introduction to the approach to satellite data handling system design for architectural-layer-driven verification

#### 3.1.1. Goal of the approach

The goal of the approach is to reduce development risks and shorten unexpected iteration testing processes. To achieve the goal, two types of architecture are to be introduced, and called as “horizontal” architecture and “vertical” architecture. “Vertical” architecture supports description of upper-layers interfaces, which cannot be done in horizontal architecture. Horizontal architecture is designed in operational domain and systems domain, including operational design, functional design, and physical design. Vertical architecture is designed with multiple layer view (vertical view), showing virtual system interfaces in some upper layers in open system interconnection model. In case of data handling system, more data interfaces are added to horizontal architecture by vertical architecture.

Since more interfaces can be identified in the architecture, they can be verified at low-level testing before system-level testing, as it is illustrated in Figure 20. This helps to shorten development iteration process, and reduce development risks by confirming upper-layers interfaces in early phases.

#### 3.1.2. Detailed approach

The approach consists of two parts including design (left side of V-model) and verification (right side of V-model) as in Figure 34.

##### 3.1.2.1. Design of data handling system

###### a) Design of horizontal architecture of data handling system

Horizontal architecture is called in order to distinguish it from vertical architecture, which will be described later. Two types of architecture are designed from different viewpoints. Horizontal architecture of satellite data handling system is designed based on Department of Defense Architecture Framework 1.0 (DoDAF1.0) using operational view and systems view in 2 domains including operational architecture domain, system architecture domain. Operational View (OV) and Systems View (SV) of DoDAF1.0

---

support design of operational activities, functionality and physical system of satellite data handling system, and identification of relationship among them.

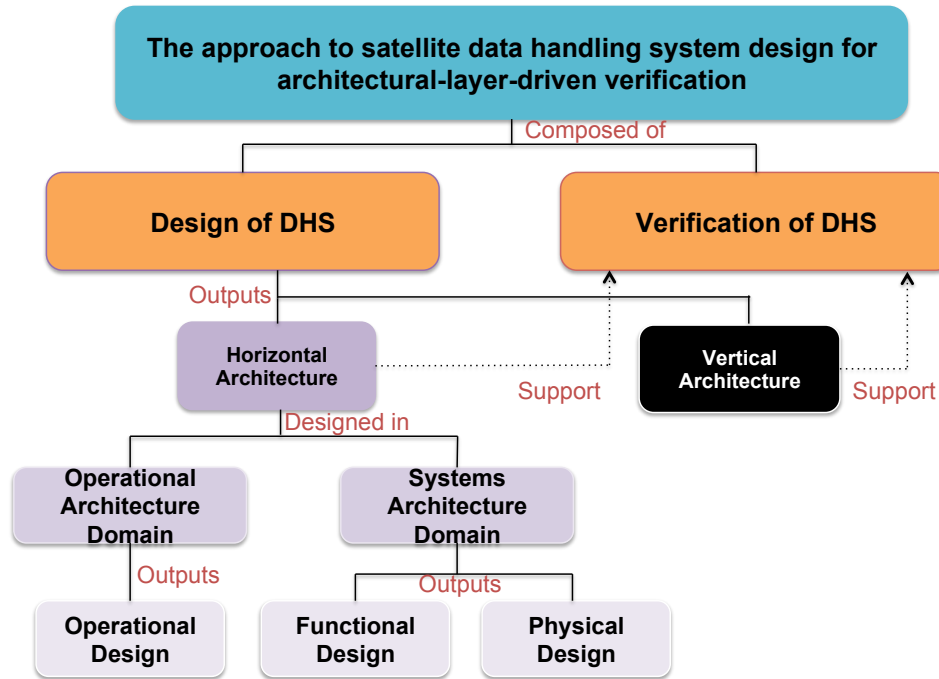


Figure 34. Introduction to the approach to satellite data handling system design for architectural-layer-driven verification

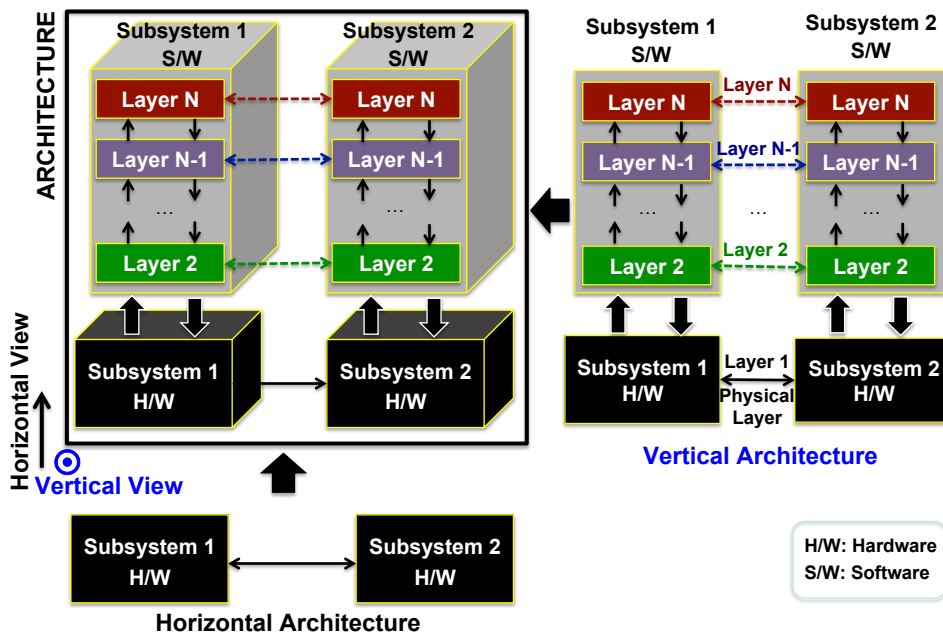


Figure 35. Basis of the approach

Operational activities affect the way that data handling system performs, thus, giving some insights of data interfaces between the satellite and the ground segment. While functional and physical design clarify how all the subsystems of DHS interact

with each other, and from that physical links and interfaces can be determined.

Basis of the approach is that 3D-image of architecture of a system can be built by looking from horizontal view to obtain horizontal architecture, and from vertical view to obtain vertical architecture Figure 35.

The relationship between operational domain and functional domain is shown in Figure 36. The horizontal architecture consists of operational design, functional design, and physical design in operational architecture domain, and systems architecture domain. In operational domain, operational activities are given based on concept of operations and capability of the system. These activities are performed and assigned to specific operational nodes, which are a part of horizontal architecture. Operational activities and operation nodes are implemented by functions and systems present in systems architecture domain. The systems are designed to perform the functions that are built from system requirements.

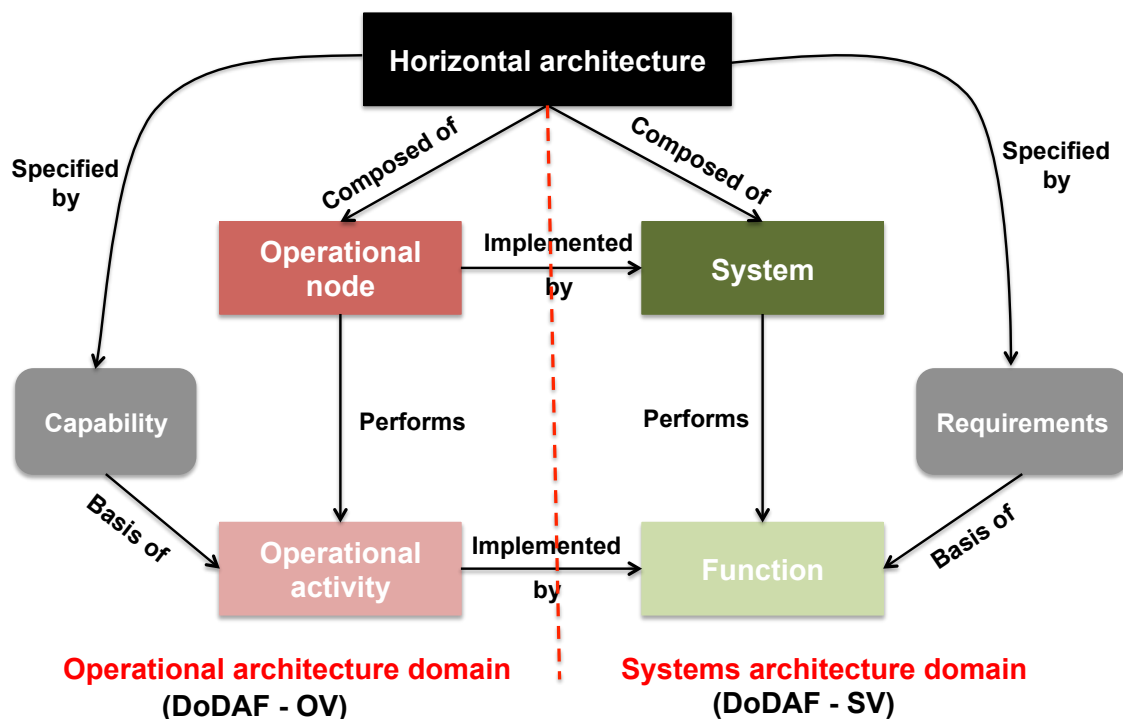


Figure 36. Relationship between operation architecture domain and systems architecture domain

#### b) Design of vertical architecture of data handling system

In this research, one new type of architecture is introduced, called “vertical” architecture. The vertical architecture is designed based on open system interconnection (OSI) model which is illustrated in Figure 37, in order to identify more interfaces in

addition to those which are described in the horizontal architecture. The purpose of designing vertical architecture is to introduce more upper-layers data interfaces in system architecture which allows to verify those interfaces at low-level testing, instead of waiting until system-level test. The way of design and test helps to shorten unexpected iteration development processes of data handling system, and leads to reduce development risks of increasing development cost and time.

Three layers of interfaces are considered in data handling system as in Figure 38, including physical layer, data link layer, and application layer. Data is transmitted between the satellite and the ground segment through antennas system. To handle data, there are some onboard computers in the satellite and a mission control center on the ground. The satellite onboard computers are used to implement ADCS software which processes TLM collected from ADCS components (gyro, star tracker, sun sensor, reaction wheel, magnetic torque, etc.) by utilizing ADCS algorithms, and to implement C&DH software which manage and control all the tasks, processes occurred in the satellite, as well as implementing mission data processing software which processes raw mission data received from mission instruments. To encode & decode the data, CCSDS (the Consultative Committee for Space Data Systems) transport services in the satellite and the ground segment are necessary. These services apply CCSDS standards & protocols to create data frames and data packets. The CCSDS services embedded in the satellite and ground segment must be compatible with each other. The following is description of 3 layers introduced in vertical architecture.

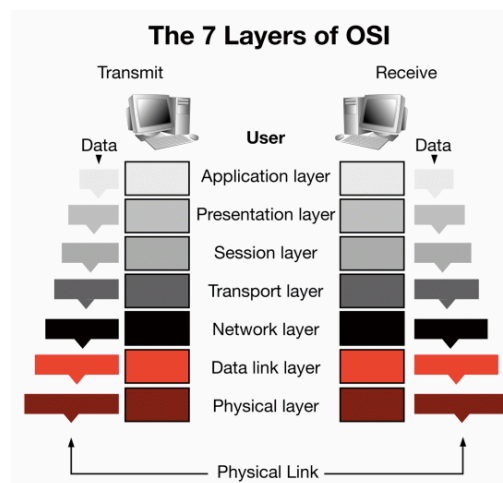


Figure 37. Open System Interconnection Model

**Layer 1 (Physical layer):** The physical layer defines data link between satellite

antennas and ground antennas. The physical layer has the following major functions:

- This layer defines electrical and physical specifications of data connection, and identifies relationship among antennas and radio frequency.
- This layer is responsible for transmission and reception of raw data using radio frequency.
- This layer defines transmission mode, such as, simplex, half duplex, full duplex, to name a few.
- This layer defines network topology, such as bus, mesh, ring, etc.

In satellite data handling system, physical layer represents radio frequency (RF) link, which carries transmission data, between the satellite antennas and ground station antennas. Types of these antennas are used based on operational frequencies of electromagnetic wave carrier.

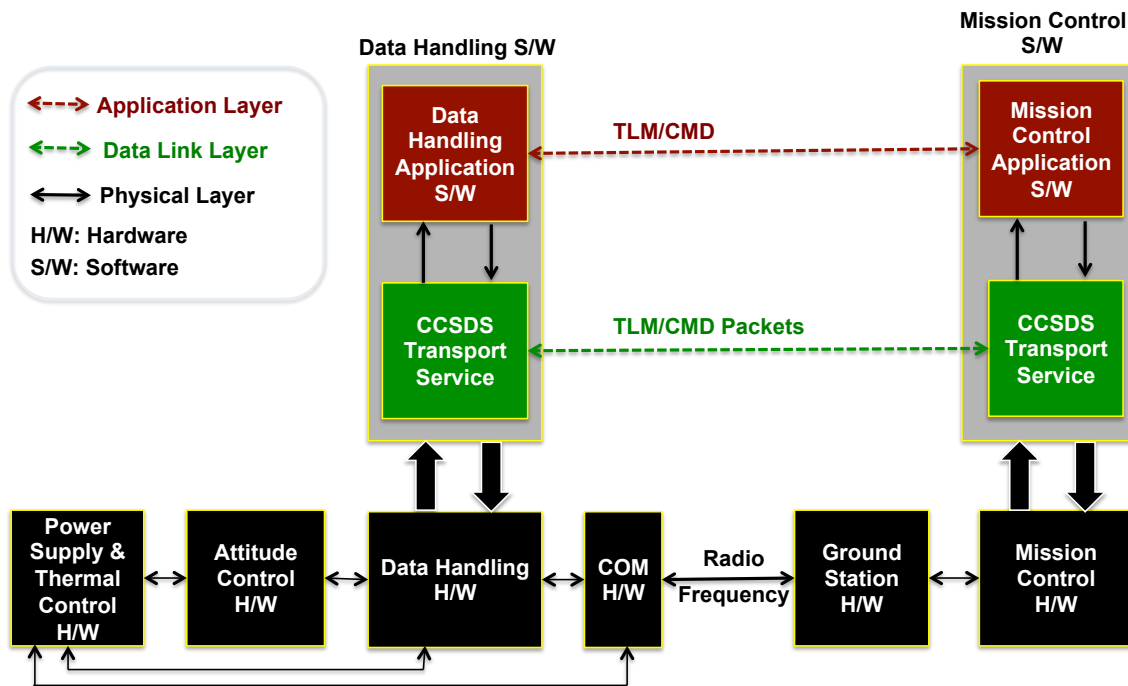


Figure 38. An example of vertical architecture of a satellite data handling system

**Layer 2 (Data link layer):** The data link layer provides node-to-node data transfer (link) between two connected devices in space and on ground. It can also detect and possibly correct may-occur errors in the physical layer. It defines the protocol to establish and terminate a connection between two devices, and defines the protocol for flow control between them.

In DHS, the data link layer describes the interface between 2 CCSDS transport

services in the satellite and the ground segment. The interface contains TLM/ CMD packet information.

**Layer 3 (Application layer):** The application layer is the layer close to the end user, in other words, the application layer and the user interact directly with onboard software which is embedded in onboard computer. The application layer functions typically include identification of communication partners, determination of resource availability, and synchronization of communication. This layer supports application and end-user processes.

Application layer of satellite data handling system represents interfaces between the satellite and the mission control center, from the sources of data in onboard software to the application ground station software where data are analyzed, displayed, and processed.

### 3.2.1.2. Verification of data handling system

Verification testing configuration is built based on interfaces identified in horizontal architecture, and vertical architecture design. The introduction of a new vertical design in the approach allows to verify upper-layers data interfaces at low-level testing, thus, reducing development risks of increasing development cost and time, as unexpected iteration development iteration processes are shortened. The verification of data handling system consists of verification of TLM/ CMD, verification of mission data, and verification of radio frequency link. Each kind of configuration is described in more detail as bellows.

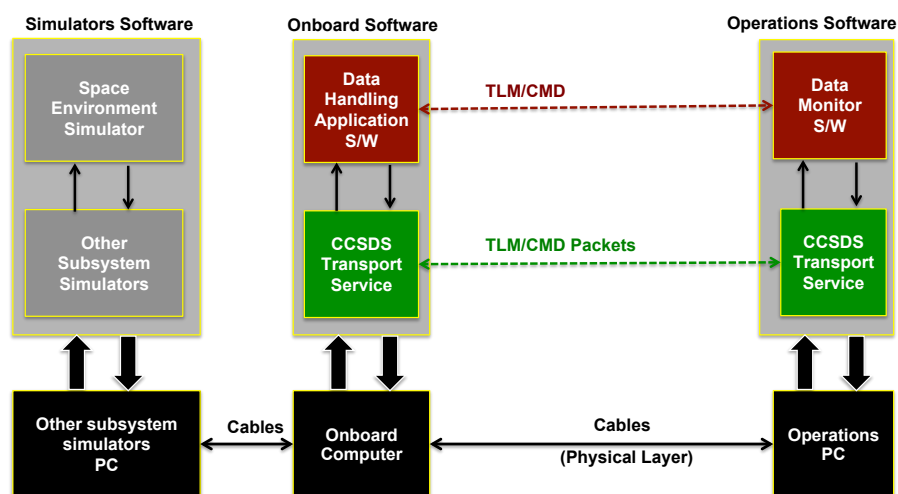


Figure 39. Verification of Telemetry/ Command

**Verification of Telemetry/ Command:** Input data from a bus component computer

play as data collected from satellite components, for example, data from gyro, non-spin sun aspect sensors, reaction wheels, star tracker, cameras, temperature sensors, etc. The input data go to onboard computer, which embeds ADCS onboard software and C&DH onboard software. The data are processed, encrypted, header-attached, and packed, then sent to operations computers that embed ground CCSDS transport service. In the ground segment, the data packets are decrypted, interpreted, processed, and displayed in display computer. It is a similar case for command, which are sent from operations computer to the onboard computer.

Verification of telemetry/ command can be done with a simple testing configuration as in Figure 39.

**Verification of mission data:** It is the same case with verification of telemetry/ command, the only difference lies at that the mission data go to science data handling unit instead of onboard computer. The verification of mission data is illustrated in Figure 40.

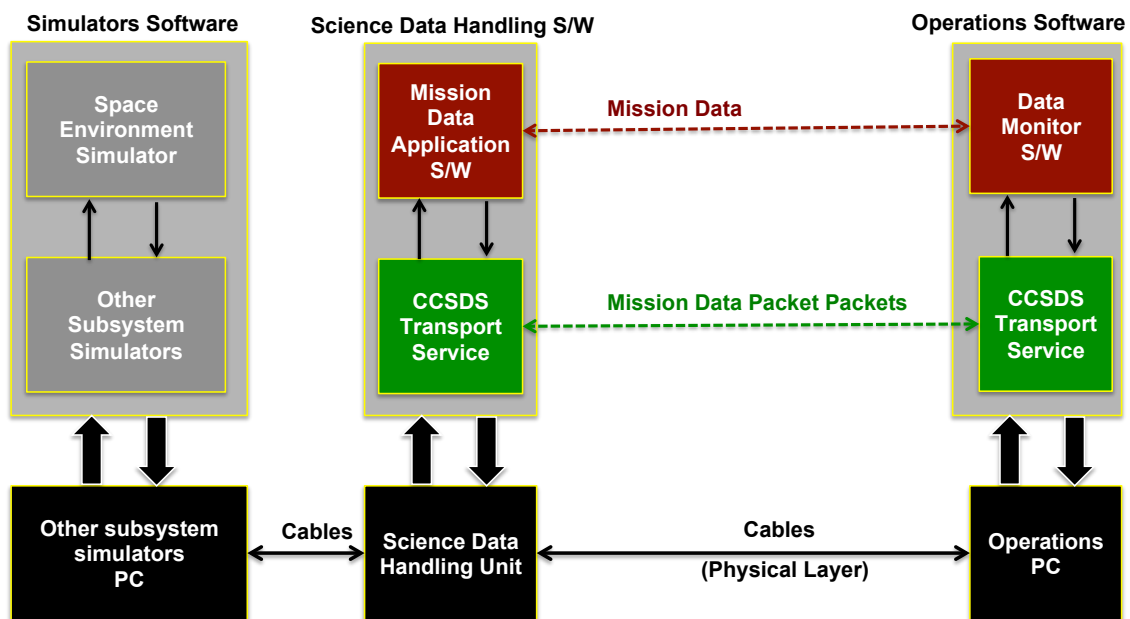


Figure 40. Verification of Mission Data

**Verification of physical layer (radio frequency):** Radio frequency layer can be verified by 2 ways. The first way is done, by connecting onboard computer of the satellite and operations computer by coaxial cable, as it is illustrated in Figure 41. In the second way, verification of radio frequency is done in end-to-end test.

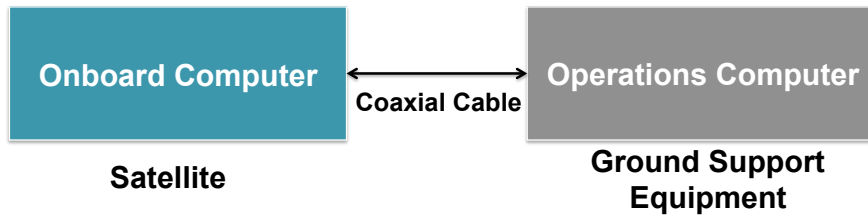


Figure 41. Verification of RF link by coaxial cable connection

Coaxial cable is a type of cable that has an inner conductor surrounded by a tubular insulating layer, surrounded by a tubular conducting shield. Coaxial cable differs from other shielded cable used for carrying lower-frequency signals, in that the dimensions of the cable are controlled to give a precise, constant conductor spacing, which is needed for it to function efficiently as a transmission line <sup>[26]</sup>. Coaxial cable is used as a transmission line for radio frequency signals. One advantage of coaxial over other types of radio transmission line is that in an ideal coaxial cable the electromagnetic field carrying the signal exists only in the space between the inner and outer conductors. Coaxial cable can protect the signal from external electromagnetic interference.

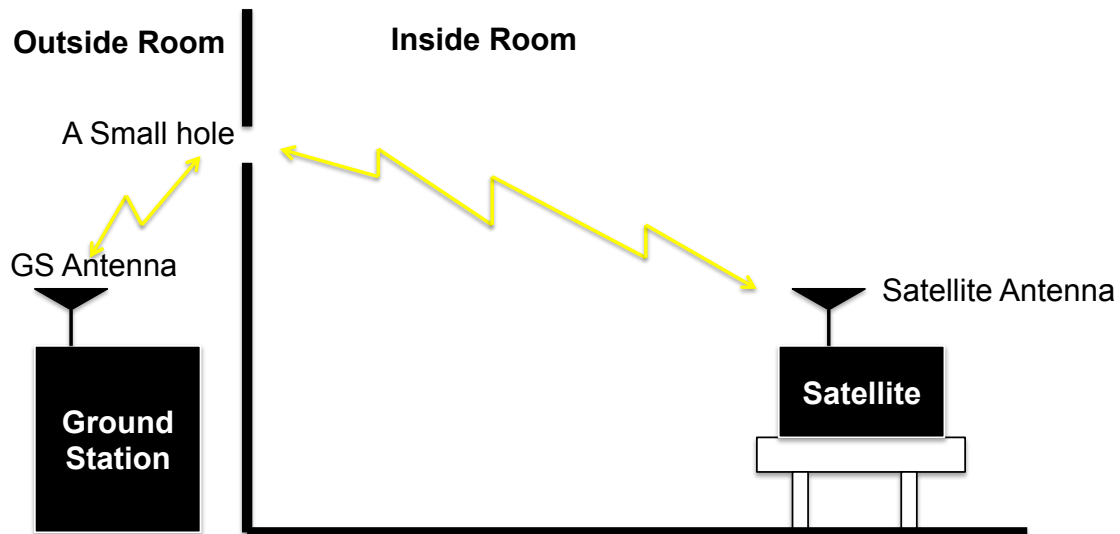


Figure 42. End-to-end testing configuration for verification of RF link

In end-to-end testing configuration for verification of RF link, illustrated in Figure 42, a satellite with its antenna is placed in a room with a small hole. Outside the room, the ground station and ground station antenna are arranged. This testing configuration allows testing communication between ground station antenna and satellite antenna.

### ***3.1.3. Introduction to Department of Defense Architecture Framework (DoDAF)***

The Department of Defense (DoD) Architecture Framework (DoDAF), version 1.0, was published in 2004. It defines a common approach for DoD architecture description development, presentation, and integration for both war-fighting operations and business operations and processes. The Framework is intended to ensure that architecture descriptions can be compared and related across organizational boundaries, including joint and multinational boundaries <sup>[24]</sup>.

An architecture description is a representation of a defined domain, as of a current or future point in time, in terms of its constituent parts, what those parts do, how the parts relate to each other and to the environment, and the rules and constraints governing them. Within the DoDAF, architectures are described in terms of three views: Operational View (OV), Systems View (SV), and Technical Standards View (TV). An architecture description is composed of architecture products that are interrelated within each view and are interrelated across views. Architecture products are those graphical, textual, and tabular items that are developed in the course of gathering architecture data, identifying their composition into related architecture components or composites, and modeling the relationships among those composites to describe characteristics pertinent to the architecture's intended use <sup>[24]</sup>.

The term architecture is generally used both to refer to an architecture description and an architecture implementation. An architecture description is a representation of a current or postulated real-world configuration of resources, rules, and relationships <sup>[24]</sup>.

In this research, some of operational view and systems view are selected to design horizontal architecture of satellite data handling system in operational domain, functional domain, and physical domain. The following is a brief description of those selected view.

#### ***3.1.3.1. Operational view***

An Operational View describes the tasks and activities, operational elements, and information exchanges required to conduct operations. A pure OV does not depend on materiel and technology. However, operations and their relationships may be influenced by new technologies such as collaboration technology, where process improvements are in practice before policy can reflect the new procedures. Three of operational view including OV-1, OV-2, and OV-5 are selected to design operational activities of

---

satellite data handling system. Table 4 describes operational view in DoDAF.

a) OV-1 (High-level operational concept graphic)

The High-Level Operational Concept Graphic describes a mission and highlights main operational nodes that will be defined later, and interesting or unique aspects of operations. It provides a description of the interactions between the subject architecture and its environment, and between the architecture and external systems. A textual description accompanying the graphic is crucial as graphics alone are not sufficient for capturing the necessary architecture data <sup>[24]</sup>.

Purpose of OV-1 is to provide a quick, high-level description of what the architecture is supposed to do, and how it is supposed to do it.

Detailed Description: OV-1 consists of a graphical executive summary for a given architecture with accompanying text. The product identifies the mission/domain covered in the architecture and the viewpoint reflected in the architecture. OV-1 should convey what the architecture is about and an idea of the players and operations involved.

**Table 4. Operational view in DoDAF 1.0 <sup>[24]</sup>**

|                    |       |   |   |
|--------------------|-------|---|---|
| <b>Operational</b> | OV-1  | High-Level Operational Concept Graphic    | High-level graphical/textual description of operational concept   |
| <b>Operational</b> | OV-2  | Operational Node Connectivity Description | Operational nodes, connectivity, and information exchange needlines between nodes   |
| <b>Operational</b> | OV-3  | Operational Information Exchange Matrix   | Information exchanged between nodes and the relevant attributes of that exchange  |
| <b>Operational</b> | OV-4  | Organizational Relationships Chart        | Organizational, role, or other relationships among organizations  |
| <b>Operational</b> | OV-5  | Operational Activity Model                | Capabilities, operational activities, relationships among activities, inputs, and outputs; overlays can show cost, performing nodes, or other pertinent information |
| <b>Operational</b> | OV-6a | Operational Rules Model                   | One of three products used to describe operational activity—identifies business rules that constrain operation  |
| <b>Operational</b> | OV-6b | Operational State Transition Description  | One of three products used to describe operational activity—identifies business process responses to events   |
| <b>Operational</b> | OV-6c | Operational Event-Trace Description       | One of three products used to describe operational activity—traces actions in a scenario or sequence of events  |
| <b>Operational</b> | OV-7  | Logical Data Model                        | Documentation of the system data requirements and structural business process rules of the Operational View   |

b) OV-2 (Operational node connectivity description)

The Operational Node Connectivity Description graphically depicts the operational nodes with needlines between those nodes that indicate a need to exchange information. The graphic includes internal operational nodes as well as external nodes <sup>[24]</sup>.

Purpose of OV-2 is intended to track the need to exchange information from

specific operational nodes to others.

**Detailed Description:** The main features of this product are the operational nodes and the needlines between them that indicate a need to exchange information. The product indicates the key players and the interactions necessary to conduct the corresponding operational activities of OV-5.

**Operational Nodes:** An operational node is an element of the operational architecture that produces, consumes, or processes information.

**Needlines:** A needline documents the requirement to exchange information between nodes. The needline does not indicate how the information transfer is implemented. Needlines are represented by arrows, and arrows on the diagram represent needlines only.

OV-2 should also illustrate needs to exchange information between operational nodes and external nodes.

**Operational Activities:** The operational activities performed by a given node may be listed on the graphic, if space permits.

#### c) OV-5 (Operational activity model)

The Operational Activity Model describes the operations that are normally conducted in the course of achieving a mission or a business goal. It describes capabilities, operational activities (or tasks), input and output (I/O) flows between activities, and I/O flows to/from activities that are outside of the scope of the architecture <sup>[24]</sup>.

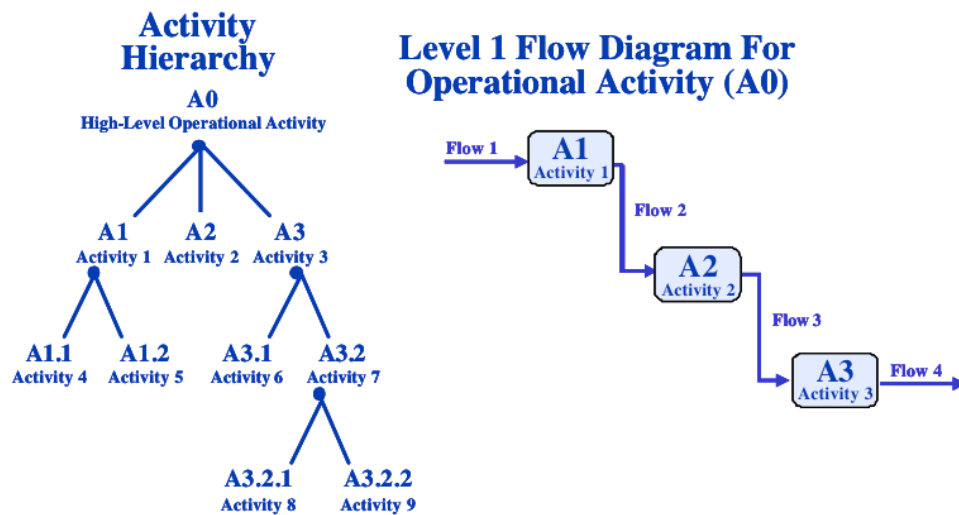


Figure 43. Operational activity hierarchy chart and operation activity diagram <sup>[24]</sup>

Purposes of OV-5 are to

- Clearly delineate lines of responsibility for activities when coupled with OV-2
- Uncover unnecessary operational activity redundancy
- Make decisions about streamlining, combining, or omitting activities
- Define or flag issues, opportunities, or operational activities and their interactions (information flows among the activities) that need to be scrutinized further
- Provide a necessary foundation for depicting activity sequencing and timing in OV-6a, OV-6b, and OV-6c

Detailed Description: OV-5 describes capabilities, operational activities (or tasks), I/O flows between activities, and I/O flows to/from activities that are outside the scope of the architecture.

OV-5 graphic may include a hierarchy chart of the activities, which is described in Figure 43. A hierarchy chart helps provide an overall picture of the activities involved and a quick reference for navigating the OV-5 I/O flow model.

### 3.1.3.2. Systems view

The Systems View is a set of graphical and textual products that describe systems and interconnections providing for, or supporting, DoD functions. SV products focus on specific physical systems with specific physical (geographical) locations. The

relationship between architecture data elements across the SV to the Operational View (OV) can be exemplified, as systems are procured and fielded to support organizations and their operations <sup>[24]</sup>. All the systems views are shown in Table 5.

Physical and functional design of architecture of satellite data handling system are done based on some systems views appropriately selected from table 3 including SV-1, SV-4, and SV-5.

a) SV-1 (Systems interface description)

SV-1 describes nodes and the systems geographical location of these nodes to support organizations/human roles represented by operational nodes defined in OV-2. SV-1 also identifies the interfaces among systems and systems nodes.

Purpose of SV-1: Identification of systems nodes and systems that support operational nodes.

Detailed Description: SV-1 connects OV and SV by depicting the assignments of systems and systems nodes to the operational nodes. OV-2 represents the operational nodes, while SV-1 represents the systems nodes and localization of systems in systems nodes. In addition, SV-1 addresses system interfaces. An interface represents one or more communications paths among systems nodes or among systems.

b) SV-4 (Systems functionality description)

SV-4 depicts system functions and system functional hierarchies, and the system data flows among them.

Purpose of SV-4: including (i) development of a clear description of the necessary system data flows between input and output respectively consumed and produced by each subsystem, (ii) Assurance of completeness of the functional connectivity, and (iii) Assurance of appropriate level of detail of the functional decomposition.

Table 5. Systems view in DoDAF 1.0 <sup>[24]</sup>

|                |        |  |   |
|----------------|--------|--|---|
| <b>Systems</b> | SV-1   | Systems Interface Description                                | Identification of systems nodes, systems, and system items and their interconnections, within and between nodes   |
| <b>Systems</b> | SV-2   | Systems Communications Description                           | Systems nodes, systems, and system items, and their related communications lay-downs  |
| <b>Systems</b> | SV-3   | Systems-Systems Matrix                                       | Relationships among systems in a given architecture; can be designed to show relationships of interest, e.g., system-type interfaces, planned vs. existing interfaces, etc.         |
| <b>Systems</b> | SV-4   | Systems Functionality Description                            | Functions performed by systems and the system data flows among system functions   |
| <b>Systems</b> | SV-5   | Operational Activity to Systems Function Traceability Matrix | Mapping of systems back to capabilities or of system functions back to operational activities   |
| <b>Systems</b> | SV-6   | Systems Data Exchange Matrix                                 | Provides details of system data elements being exchanged between systems and the attributes of that exchange  |
| <b>Systems</b> | SV-7   | Systems Performance Parameters Matrix                        | Performance characteristics of Systems View elements for the appropriate time frame(s)  |
| <b>Systems</b> | SV-8   | Systems Evolution Description                                | Planned incremental steps toward migrating a suite of systems to a more efficient suite, or toward evolving a current system to a future implementation                             |
| <b>Systems</b> | SV-9   | Systems Technology Forecast                                  | Emerging technologies and software/hardware products that are expected to be available in a given set of time frames and that will affect future development of the architecture    |
| <b>Systems</b> | SV-10a | Systems Rules Model  | One of three products used to describe system functionality—identifies constraints that are imposed on systems functionality due to some aspect of systems design or implementation |
| <b>Systems</b> | SV-10b | Systems State Transition Description                         | One of three products used to describe system functionality—identifies responses of a system to events  |
| <b>Systems</b> | SV-10c | Systems Event-Trace Description                              | One of three products used to describe system functionality—identifies system-specific refinements of critical sequences of events described in the Operational View                |
| <b>Systems</b> | SV-11  | Physical Schema  | Physical implementation of the Logical Data Model entities, e.g., message formats, file structures, physical schema   |

Detailed Description: SV-4 describes system functions and the flow of system data among system functions. SV-4 may focus on intra-nodal system data flow, inter-nodal system data flow, system data flow without node considerations, function to system allocations, and function to node allocations.

SV-4 may have both a hierarchy (decomposition model) shown in Figure 44 and a system data flow model shown in Figure 45. The hierarchy model represents a functional decomposition.

- c) SV-5 (Operational activity to systems function traceability matrix): The matrix has one template as described in Figure 46.

Operational Activity to Systems Function Traceability Matrix is a specification of the relationships between the set of operational activities applicable to architecture, and the set of system functions applicable to that architecture <sup>[24]</sup>.

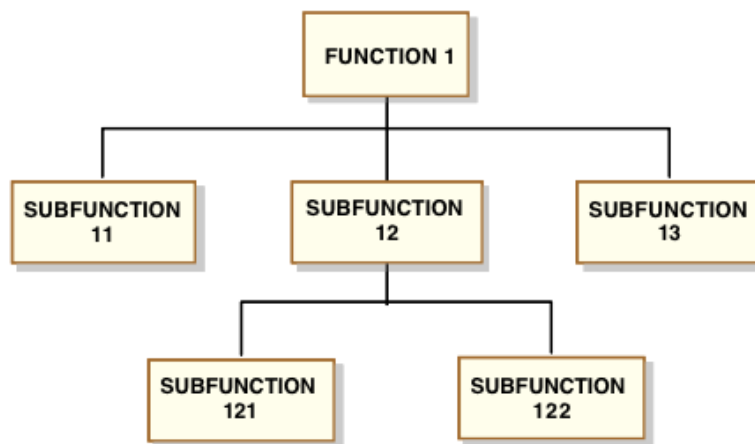


Figure 44. Decomposition model of systems functionality description (SV-4)

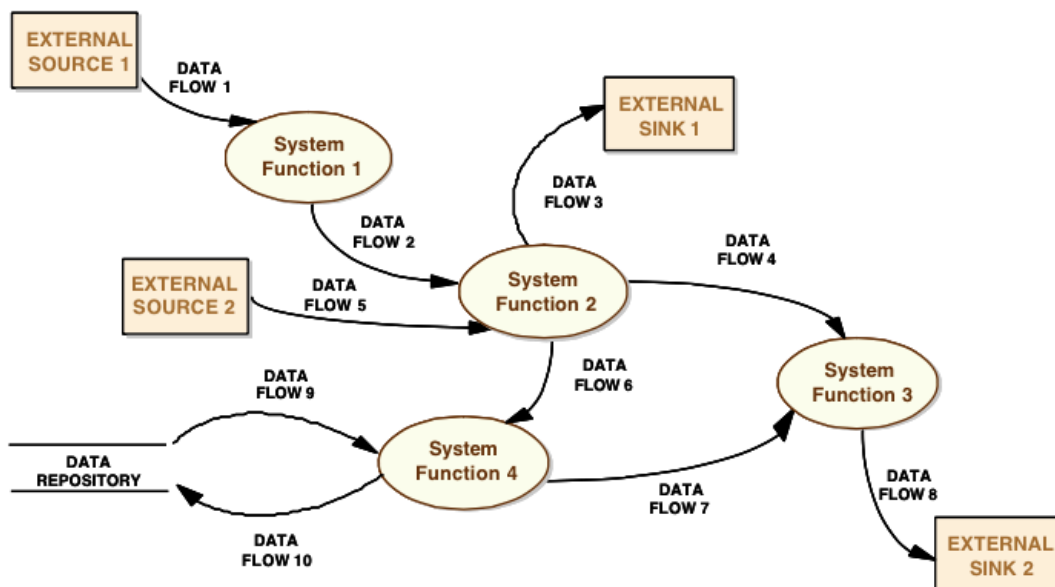


Figure 45. System data flow model of systems functionality description (SV-4)

Purpose of SV-5: representation of the mapping of operational activities to system functions. Extended SV-5 can also depict mapping of capabilities to operational activities, operational activities to system functions, system functions to systems.

Detailed Description: The term of OV and SV are used for activities and functions of which tasks are to be performed, to accept inputs, and to develop outputs. The distinction is created by the fact that system functions are executed by automated systems, while operational activities describe business operations that may be conducted by humans, automated systems, or both.

| System Functions |  | Operational Activities |        |      |        |        |        |      |      |        |        |        |        |      |      |      |        |
|------------------|--|------------------------|--------|------|--------|--------|--------|------|------|--------|--------|--------|--------|------|------|------|--------|
|                  |  | 3.11                   | 3.11.3 | 3.12 | 3.12.1 | 3.12.2 | 3.12.3 | 3.13 | 3.14 | 3.14.1 | 3.14.2 | 3.14.3 | 3.14.4 | 3.15 | 3.16 | 3.17 | 3.17.1 |
|                  |  |                        |        |      |        |        |        |      |      |        |        |        |        |      |      |      |        |
| 1                |  | X                      |        |      |        |        |        |      |      |        |        |        |        |      |      |      |        |
| 1.1              |  |                        | X      |      |        |        |        |      |      |        |        |        |        |      |      |      |        |
| 1.1.1            |  |                        |        | X    |        |        |        |      |      |        |        |        |        |      |      |      |        |
| 1.1.1.1          |  | X                      |        |      |        |        |        |      |      |        |        |        |        |      |      |      |        |
| 1.1.1.2          |  |                        |        |      | X      |        |        |      |      |        |        |        |        |      |      |      |        |
| 1.1.1.3          |  |                        |        |      |        | X      |        |      |      |        |        |        |        |      |      |      |        |
| 1.1.2            |  |                        |        |      |        |        | X      |      |      |        |        |        |        |      |      |      |        |
| 1.1.2.1          |  |                        |        | X    |        |        |        |      |      | X      |        |        |        |      |      |      |        |
| 1.1.2.2          |  |                        |        |      |        | X      |        |      |      |        |        |        |        |      |      |      |        |
| 1.1.2.3          |  |                        |        |      |        |        |        | X    |      |        |        |        |        |      |      |      |        |
| 1.1.3            |  |                        |        |      |        |        |        |      |      |        | X      |        |        |      |      |      |        |
| 1.1.3.1          |  |                        |        |      |        |        |        |      |      |        |        |        | X      |      |      |      |        |
| 1.1.3.2          |  |                        |        |      |        |        |        |      | X    |        |        |        |        |      |      |      |        |
| 1.1.3.3          |  |                        |        |      |        |        |        |      |      |        |        |        |        |      | X    |      |        |
| 1.1.3.4          |  |                        |        |      |        |        |        |      |      |        |        |        |        |      | X    |      |        |

Figure 46. Operational activity to systems function traceability matrix (SV-5)

### 3.2. Originality of the approach

This research is introducing an approach to satellite data handling system design for architectural-layer-driven verification. The approach in this research is developed to accomplish the goal of reducing development risks and unexpected iteration testing processes. The development risks and iteration testing processes can be reduced, by confirming upper-layers interfaces at low-level testing, instead of system-level testing, as it is described in Figure 20. Verification of upper-layers interfaces can be done via architecture design process, which outputs horizontal architecture and vertical architecture. While before, upper-layers interfaces was identified and verified based on experience of development engineers without any explicit systematic approaches. The following points clarify the originality of the research compared to previous ones.

- Addition of vertical view to design vertical architecture: Before in previous approaches, architecture is designed based on operational, functional & physical

properties of a system as horizontal architecture in the proposed approach, and vertical architecture was not defined. The proposed approach supports in building architecture from horizontal view and additional vertical view, which allows to built 3D-image of architecture.

- The proposed approach offers a systematic classification of system architecture into vertical architecture and horizontal architecture. In space industry, before experienced engineers identified and tested upper-layers interfaces based on their experiences without any explicit way. With this approach, satellite development engineers can fully define and test data handling system in systematic way. The systematic way of designing and testing data handling system shows a great advantage when applying to a complex system, such as a data relay satellite system where more than two satellites, and more than two ground segments, connect to each other to share data. The data relay satellite system is illustrated in Figure 47.
- The vertical architecture in the proposed approach supports in identifying missed upper-layer interfaces, which could only be defined by experienced engineers based on their experiences.

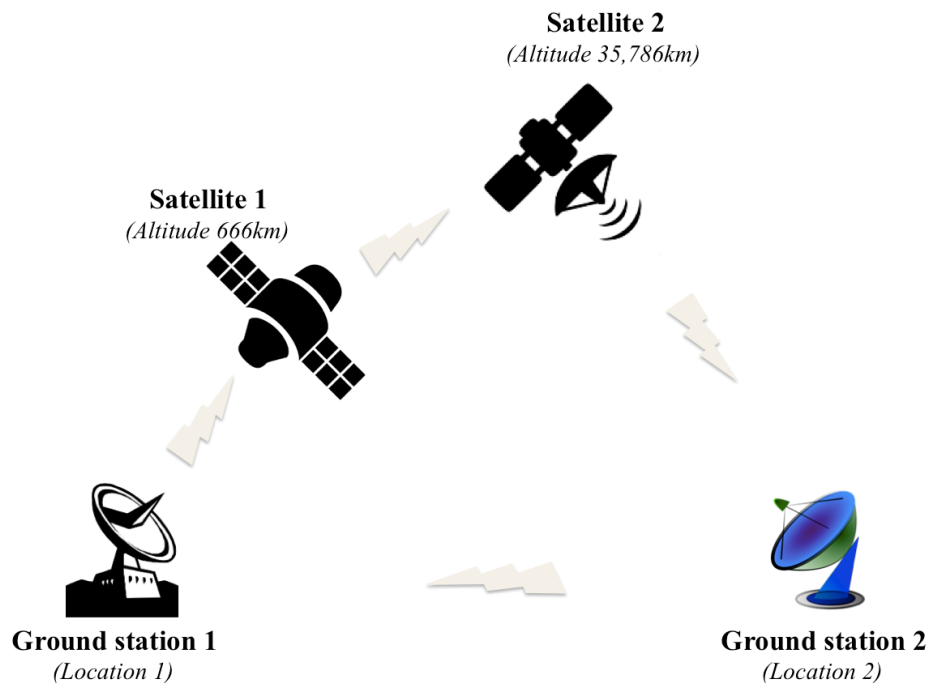


Figure 47. A data relay satellite system

## Chapter 4 Evaluation of the approach

### 4.1. Evaluation of the approach

Evaluation of the approach includes verification and validation. The verification is to be done, by applying the approach to design and verify data handling system of MicroDragon satellite that was introduced in section 1.1.3, and that of a data relay satellite system, illustrated in Figure 47, and by asking satellite specialists' opinions and comments on the approach (system) and its parts (subsystems), and architecture design of data handling system of MicroDragon satellite. The validation is to be done by confirming with satellite specialists whether the approach achieves the objective of the research.

#### *4.1.1. Applying the approach to develop DHS of MicroDragon satellite*

The approach is applied to design architecture and verify data handling system of MicroDragon satellite. The output includes horizontal and vertical architecture of DHS, together with test results obtained from TableSat activities at Tokyo University. The designing and verifying processes of data handling system of MicroDragon satellite is described in detail in the next chapter.

#### *4.1.2. Applying the approach to develop DHS of a data relay satellite system*

The data relay satellite system, introduced in this thesis, consists of a low-earth orbit (LEO) satellite, and a geostationary (GEO) satellite, together with their ground segments. The system is illustrated in Figure 47. The approach is applied to this system in order to emphasize the advantage of this approach in term of designing and verifying data handling system of a complex satellite system.

In this thesis, detailed designing process is not shown, only results and testing configuration of data handling system of the data delay satellite system are to be described.

#### *4.1.3. Interviewing satellite specialists*

Interviews will be done with five satellite specialists from different space backgrounds in Japan. Table 6 shows detailed information of the specialists.

Table 6. List of interviewees for evaluation of the approach

| Interviewee ID        | Description   |
|-----------------------|---|
| The interviewee No.1  | He has occupied himself with space industry for more than 30 years. Currently, he owns a professorship at Tokyo university. He has engaged as a project manager in many satellite development projects for Microsatellites, deep-space satellites, to name a few.   |
| The interviewee No.2  | He worked for Tokyo university before having a senior project assistant professorship at Keio SDM. He has taken part in several satellite development projects as a hardware manager, particularly, as an EPS (Electrical Power Supply subsystem) specialist.   |
| The interviewee No.3  | He had worked for one satellite company in Japan for more than 20 years before leaving for Keio SDM to work as a professor. He specializes in system engineering, and logical thinking.   |
| The interviewee No. 4 | He is currently working for Keio SDM as a professor. He engaged in one microsatellite development project in Japan some years ago. He specializes in system engineering.  |
| The interviewee No. 5 | He worked at space companies in Japan for more than 10 years before having experience of 7 years of working for National Aerospace Laboratory of Japan, and 2 years of working for Japan Aerospace and Exploration Agency. Currently, he is working as project professor at Keio SDM. He has participated as a project manager and a system manager in several satellite projects where variety of satellites from 50Kg to 2.5 Ton, have been developed. He also worked as a project manager in supersonic transportation system development project. |

Interview questions focus on understandability (question 1-6), usability (question 7-9) and effectiveness of the approach at both levels including system-level (the

approach) and subsystem-level (parts of the approach) (question 10-15) for verification, and validation (question 16-18). Table 7 shows in detail the questionnaire.

**Table 7. List of questions for evaluation of the approach**

|   |                 |   |
|---|-----------------|---|
| 1 | <b>Question</b> | How well do you understand about the satellite data handling system explained and mentioned in the thesis?  |
|   | <b>Answer</b>   | A. I totally understand (90% - 100%)<br>B. I almost understand (70% - 90%)<br>C. I fairly understand (40% - 70%)<br>D. I do not understand (<40%)<br>E. Other answers |
| 2 | <b>Question</b> | How well do you understand about the overall approach to satellite data handling system design for architectural-layer-driven verification?                           |
|   | <b>Answer</b>   | A. I totally understand (90% - 100%)<br>B. I almost understand (70% - 90%)<br>C. I fairly understand (40% - 70%)<br>D. I do not understand (<40%)<br>E. Other answers |
| 3 | <b>Question</b> | How well do you understand about horizontal and vertical architecture mentioned in this approach?   |
|   | <b>Answer</b>   | A. I totally understand (90% - 100%)<br>B. I almost understand (70% - 90%)<br>C. I fairly understand (40% - 70%)<br>D. I do not understand (<40%)<br>E. Other answers |
| 4 | <b>Question</b> | How well do you understand about verification methods of data handling system mentioned in this approach?   |
|   | <b>Answer</b>   | A. I totally understand (90% - 100%)<br>B. I almost understand (70% - 90%)<br>C. I fairly understand (40% - 70%)<br>D. I do not understand (<40%)<br>E. Other answers |

|   |                 |   |
|---|-----------------|---|
| 5 | <b>Question</b> | How well do you know about MicroDragon satellite, such as the size, weight, mission and developers of MicroDragon satellite?  |
|   | <b>Answer</b>   | A. I totally understand (90% - 100%)<br>B. I almost understand (70% - 90%)<br>C. I fairly understand (40% - 70%)<br>D. I do not understand (<40%)<br>E. Other answers |
| 6 | <b>Question</b> | How well do you understand about TableSat activities mentioned in this research?  |
|   | <b>Answer</b>   | A. I totally understand (90% - 100%)<br>B. I almost understand (70% - 90%)<br>C. I fairly understand (40% - 70%)<br>D. I do not understand (<40%)<br>E. Other answers |
| 7 | <b>Question</b> | How easy do you think about applying this approach to design horizontal architecture of data handling system?   |
|   | <b>Answer</b>   | A. It is very easy<br>B. It is easy<br>C. It is a little bit difficult<br>D. It is very difficult<br>E. Other answers   |
| 8 | <b>Question</b> | How easy do you think about applying this approach to design vertical architecture of data handling system?   |
|   | <b>Answer</b>   | A. It is very easy<br>B. It is easy<br>C. It is a little bit difficult<br>D. It is very difficult<br>E. Other answers   |
| 9 | <b>Question</b> | How easy do you think about applying this approach to verify data handling system?  |
|   | <b>Answer</b>   | A. It is very easy<br>B. It is easy   |

|    |                 |  |
|----|-----------------|--|
|    |                 | <p>C. It is a little bit difficult</p> <p>D. It is very difficult</p> <p>E. Other answers</p>  |
| 10 | <b>Question</b> | Do you think that the approach is a good way of design of satellite data handling system?  |
|    | <b>Answer</b>   | <p>A. Yes, I think it is a very good way</p> <p>B. Yes, I think it is a good way</p> <p>C. No, I do not think it is not a good way</p> <p>D. No, I think it is not good way at all</p> <p>E. Other answers</p>   |
| 11 | <b>Question</b> | Do you think that horizontal architecture design is useful for verification of data handling system?   |
|    | <b>Answer</b>   | <p>A. Yes, I think it is very useful</p> <p>B. Yes, I think it is useful</p> <p>C. No, I do not think it is useful</p> <p>D. No, I think it is not useful at all</p> <p>E. Other answers</p>                     |
| 12 | <b>Question</b> | Do you think that vertical architecture design is useful for verification of data handling system?   |
|    | <b>Answer</b>   | <p>A. Yes, I think it is very useful</p> <p>B. Yes, I think it is useful</p> <p>C. No, I do not think it is useful</p> <p>D. No, I think it is not useful at all</p> <p>E. Other answers</p>                     |
| 13 | <b>Question</b> | Do you think that the approach is a useful way of verification of data handling system?  |
|    | <b>Answer</b>   | <p>A. Yes, I think it is a very useful way</p> <p>B. Yes, I think it is an useful way</p> <p>C. No, I do not think it is an useful way</p> <p>D. No, I think it is not an useful way</p> <p>E. Other answers</p> |
| 14 | <b>Question</b> | Do you think that horizontal architecture design of data handling  |

|    |                 |   |
|----|-----------------|---|
|    |                 | system of MicroDragon satellite is good?  |
|    | <b>Answer</b>   | A. Yes, I think it is very good<br>B. Yes, I think it is good<br>C. No, I do not think it is good<br>D. No, I think it is not good at all<br>E. Other answers   |
| 15 | <b>Question</b> | Do you think that vertical architecture design of data handling system of MicroDragon satellite is good?  |
|    | <b>Answer</b>   | A. Yes, I think it is very good<br>B. Yes, I think it is good<br>C. No, I do not think it is good<br>D. No, I think it is not good at all<br>E. Other answers   |
| 16 | <b>Question</b> | Do you think that applying this approach to design and verification of data handling system of MicroDragon satellite can reduce development risks, especially for development of a new system?                          |
|    | <b>Answer</b>   | A. Yes, I think so<br>B. No, I do not think so<br>C. Other answers  |
| 17 | <b>Question</b> | Do you think that the approach is a beneficial way of design and verification of data handling system of a complex satellite system, such as a data relay satellite system, in terms of reduction of development risks? |
|    | <b>Answer</b>   | A. Yes, I think so<br>B. No, I do not think so  |
| 18 | <b>Question</b> | Do you think that the approach has some advantages over some other approaches mentioned in this thesis, in terms of reduction of development risks?   |
|    | <b>Answer</b>   | A. Yes, I think so<br>B. No, I do not think so  |

## 4.2. Applying the approach to develop DHS of MicroDragon satellite

### 4.2.1. Architecture design of DHS of MicroDragon satellite

Architecture of DHS of MicroDragon satellite consists of horizontal and vertical architecture. The horizontal architecture describes operations, functionality, and physical design of DHS of MicroDragon satellite, where interfaces/ links in the system are represented. The vertical architecture adds some upper-layers data interfaces.

#### 4.2.1.1. Horizontal architecture design of DHS of MicroDragon satellite

The horizontal architecture is built in operational architecture domain and systems architecture domain, based on operational view OV-1 → OV-5 → OV-2, and systems view SV-4 → SV-5 → SV-1 of DoDAF.

##### a) Operational view

**OV-1** describes high-level concept of operations (ConOPs) of MicroDragon, as it is shown in Figure 48. The mission of MicroDragon (MDG) satellite is to observe color of the water in Vietnamese sea areas in order to access water quality and locate aquatic creatures.

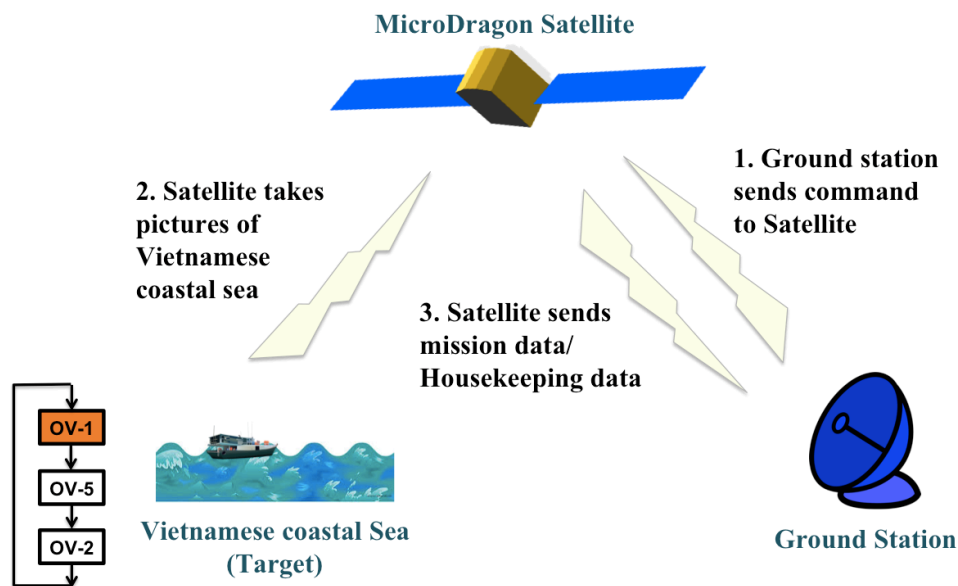


Figure 48. Concept of operations of MicroDragon satellite

The satellite has 4 cameras, when it passes over the targeted areas, it will take the pictures and store the image data in onboard computer (OBC). This process can be automatically controlled by software programs, embedded in OBC, or manually controlled by uplink command from the ground station. The ground segment located in

Hanoi of MDG satellite includes a mission control center, one X-band antenna to receive mission data, and one S-band antenna to send/receive CMD/TLM respectively. Every time when the satellite passes over its ground station, it will downlink the data including housekeeping (TLM) and mission data, and followed by processing and storing of data in the ground segment. The data downlink process can also be implemented automatically by the software, or manually by sending uplink command from the ground station.

**OV-5** (Operational activities of DHS of MDG satellite): Operational activities are defined as tasks and assignments which can normally be performed by human or machines. Based on concept of operations described in OV-1, the operational activities of DHS of MDG satellite are divided into activities in three phases including initial operations, nominal operations, and end-of-operations. Figure 49 describes broken down structure of operational activities. Figure 50 shows the flow of operational activities. Then those three activities are to be broken down to lower levels in such a way that it is able to localize each operational activity to a specific operational node, and specific functions of DHS.

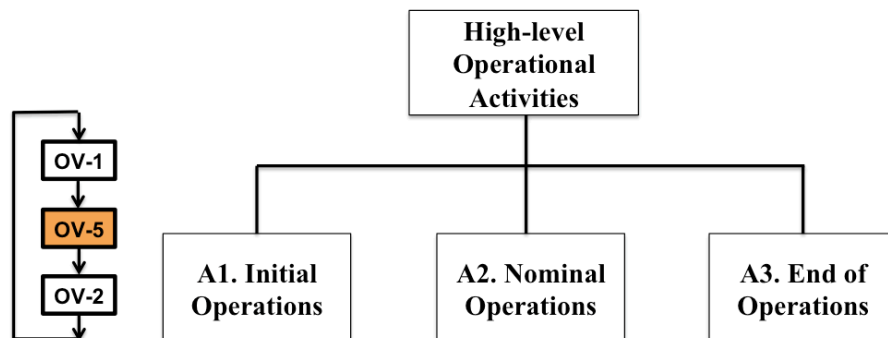


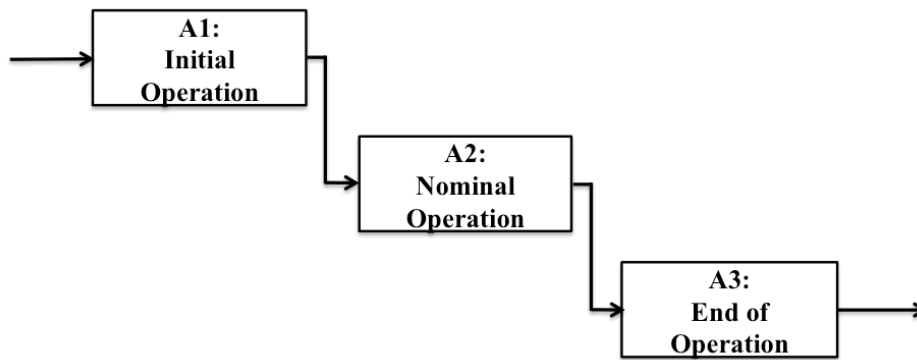
Figure 49. High-level operational activities of DHS of MicroDragon satellite

The initial operations includes activities of the satellite and the ground station in initial phase after the satellite separates from the rocket, and before the satellite goes to nominal operations or reaches to safe & stable state to perform the mission. There are some conditions to determine whether the satellite exits initial operations to nominal operations, such as success of deployment of solar array paddle, voltage of battery is bigger than requirements, or implementation of an uplink command to switch de-tumbling mode to another mode.

After exiting the initial phase, the satellite enters nominal operations when the satellite can perform the mission. Some operations modes are considered in this phase,

including spin sun acquisition mode where satellite attitude is controlled by magnetic torque, 3-axis sun acquisition mode where satellite attitude is controlled by reaction wheels, coarse earth acquisition mode where the satellite can downlink store & forward data, fine earth acquisition mode where the satellite can perform store & forward mission, and fine pointing earth where the satellite can capture pictures and downlink data.

If the satellite wants to end its lifecycle, it has to deorbit by falling off to the ground. In order to protect human and environment, the satellite has to be able to burn itself under effect of friction with the earth atmosphere. The position where the satellite may land on the earth should also be carefully calculated. Normally, satellite operators try to control the satellite to fall off in the ocean when it deorbits.



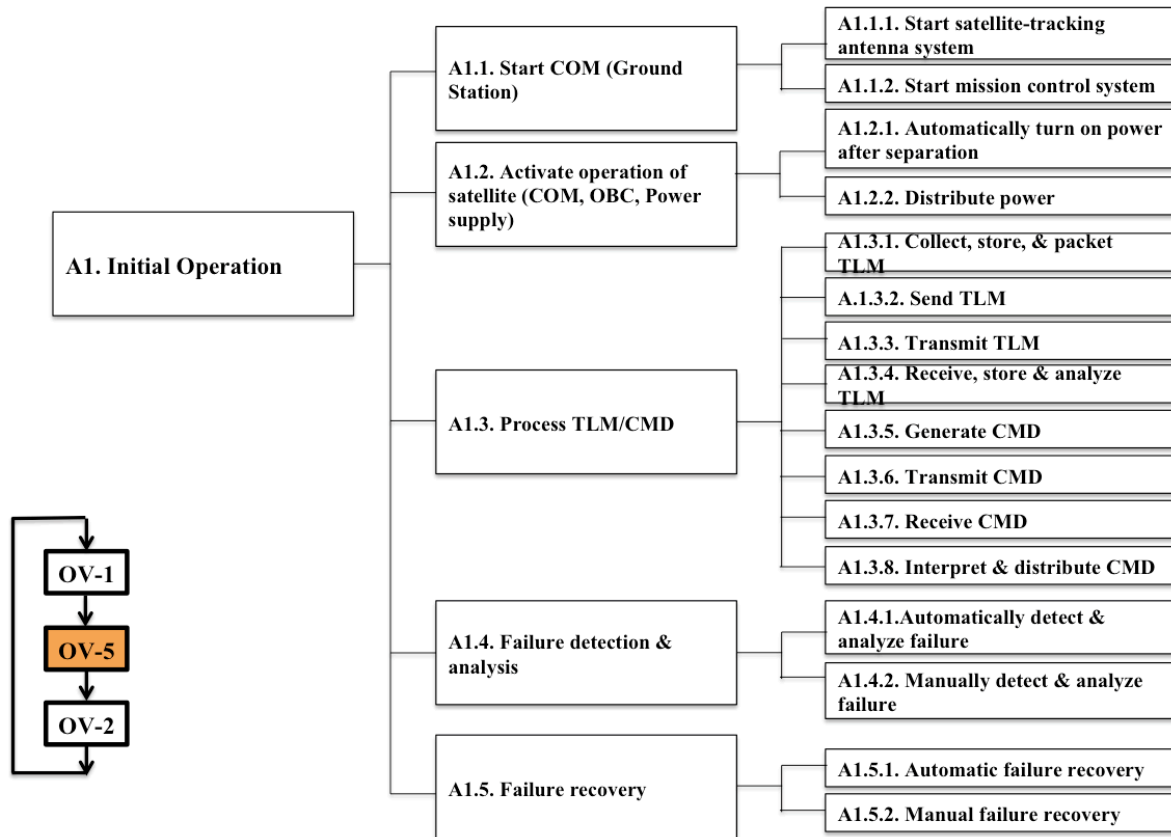
**Figure 50. High-level operational activities flow of DHS of MicroDragon satellite**

In initial operations, broken down structure and flow of operational activities are shown in Figure 51 and Figure 52. Communications at the ground station must be in readiness state, even before the separation, as it is necessary to check the health of the satellite by observing first telemetries received from the satellite, and to send first commands to operate the satellite. Initial operations phase is very critical, as it determines whether the satellite has any survival chances in space after separation from the rocket. As soon as the satellite is separated from the rocket, some important components are activated to maintain minimal operations, including onboard computer (OBC) to control and manage all the activities, power control unit (PCU) to supply the power, and S-band transponder (STRX) to send/receive TLM/CMD.

The command is generated from the mission control center, and sent to the satellite via the ground station S-band antenna and the satellite S-band antenna. This command is then sent to onboard computer to be processed and distributed to relevant satellite

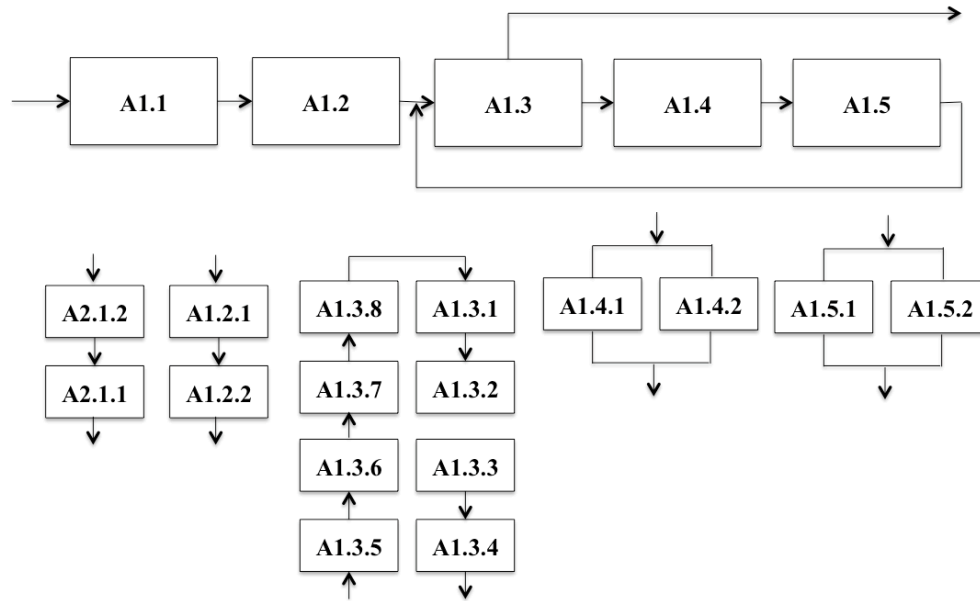
components. After receiving the command, these components will perform their dedicated tasks and functions such as turn on/ off, capture the pictures, etc.

The telemetry is collected from satellite components frequently, and sent to onboard computer. The frequency of collecting the data depends on operations control and management algorithms of C&DH software, and ADCS software, for example, telemetry can be collected and updated one time per second.



**Figure 51. Operational activities in initial operations of DHS of MicroDragon satellite (A1)**

Figure 52 shows the sequence of operational activities in initial operations of data handling system of MicroDragon satellite. At first, communication part in the ground segment should be started up (A1.1), even before the satellite separates from the rocket. This condition guarantees that the satellite can communicate with the ground station at any moments after separation. To maintain minimal operations, some satellite components should be turned on as soon as the separation (A1.2), for example onboard computer to control and manage all the processes, S-band transponder (STRX) to transmit and receive data, and power control unit (PCU) to supply electrical power.



**Figure 52. A1 operational activities flow of DHS of MicroDragon satellite**

Processing telemetry and command (A1.3) is one of most important tasks of data handling system. It provides smooth operations between the satellite and the ground segment. With telemetry received from the satellite, the mission control center can check health status of the satellite and each satellite component, as well as the information of orbit and attitude of the satellite, etc. The cycle of processing TLM/CMD can be defined from collection, storage and package of data by onboard computer (A1.3.1) → sending TLM package by satellite S-band antenna (A1.3.2) → transmission of TLM package by ground station S-band antenna (A1.3.3) → reception, storage and analysis of TLM by mission control center (A1.3.4) → generation of CMD by mission control center (A1.3.5) → transmission of CMD by the ground station S-band antenna (A1.3.6) → receiving CMD by satellite S-band antenna (A1.3.7) → processing and distributing CMD to relevant satellite components by onboard computer (A1.3.8).

In initial phase, the satellite is under fragile state, thus, failures detection and recovery (A1.4 & A1.5) are essential. For those severe failures, it is necessary to detect and recover as soon as possible by the software programs embedded in OBC. The other failures can be detected by analyzing relevant telemetries, and fixed by appropriate uplink commands. In case of occurrence of failures, detection, analysis and recovery should be done before doing any other tasks.

In nominal phase, Figure 53 and Figure 54 describe broken down structure and flow of operational activities. Communication in the ground segment can be activated or

deactivated depending on whether the ground station wants to communicate with the satellite (A2.1). The ground station can send/ receive command/ telemetry to/ from the satellite. The cycle of processing CMD (A2.2) can be defined from generation of CMD by the mission control center (A2.2.1) → transmission of CMD by the ground station S-band antenna (A2.2.2) → reception of CMD by the satellite S-band antenna (A2.2.3) → processing and distributing CMD (A2.2.4). The cycle of processing TLM (A2.3) starts from collection, storage and package of TLM by onboard computer (A2.3.1) → sending TLM by the satellite S-band antenna (A2.3.2) → transmission of TLM package by the ground station S-band antenna (A2.3.3) → reception, analysis and storage of TLM (A2.3.4).

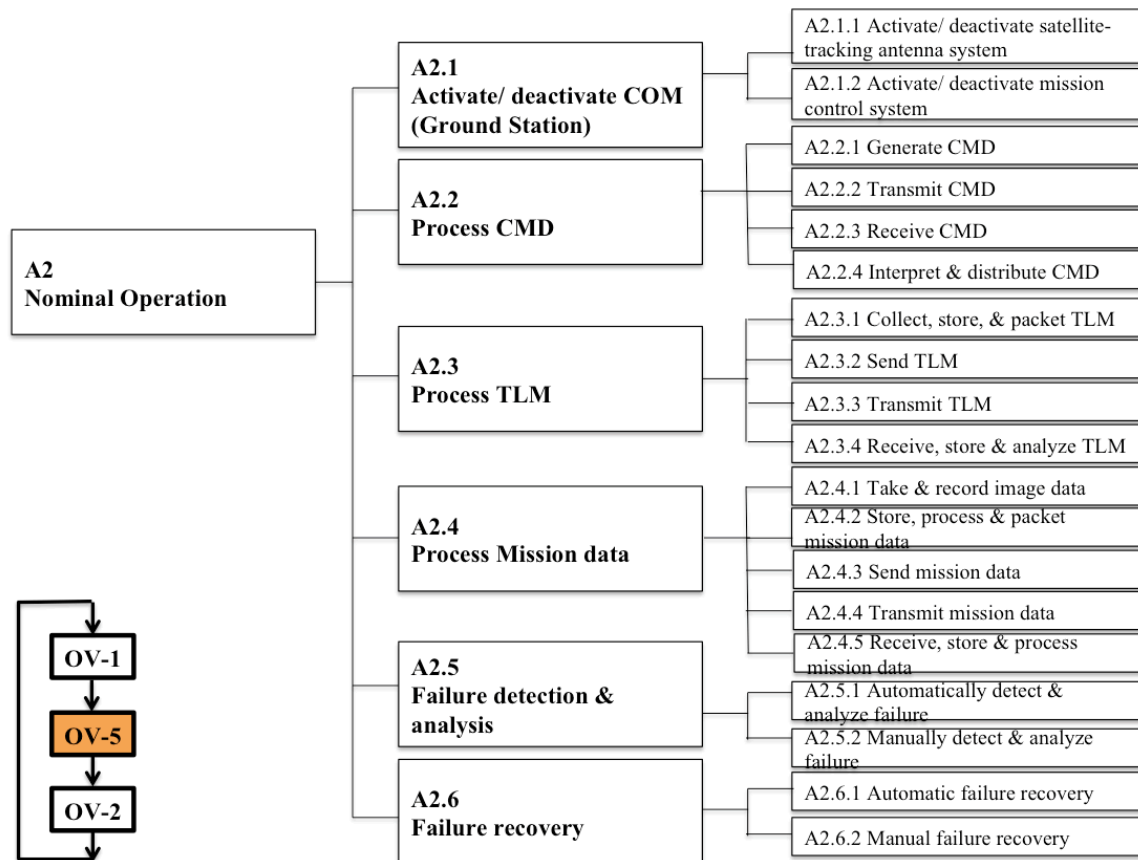


Figure 53. Operational activities in nominal phase of DHS of MicroDragon satellite (A2)

MicroDragon satellite performs the mission of capturing images of Vietnamese sea areas. The process of mission data (A2.4) starts from capturing and recording image data by cameras (A2.4.1) → storing, processing and packaging mission data by science data handling unit (A2.4.2) → sending mission data by the satellite X-band antenna (A2.4.3) → transmission of mission data by the ground station X-band antenna (A2.4.4)

→ receiving, processing and storing mission data (A2.4.5).

Similarly to initial operations, in nominal phase, failures can be detected and recovered automatically by the software or manually by relevant telemetry analysis and appropriate uplink command (A2.5 and A2.6).

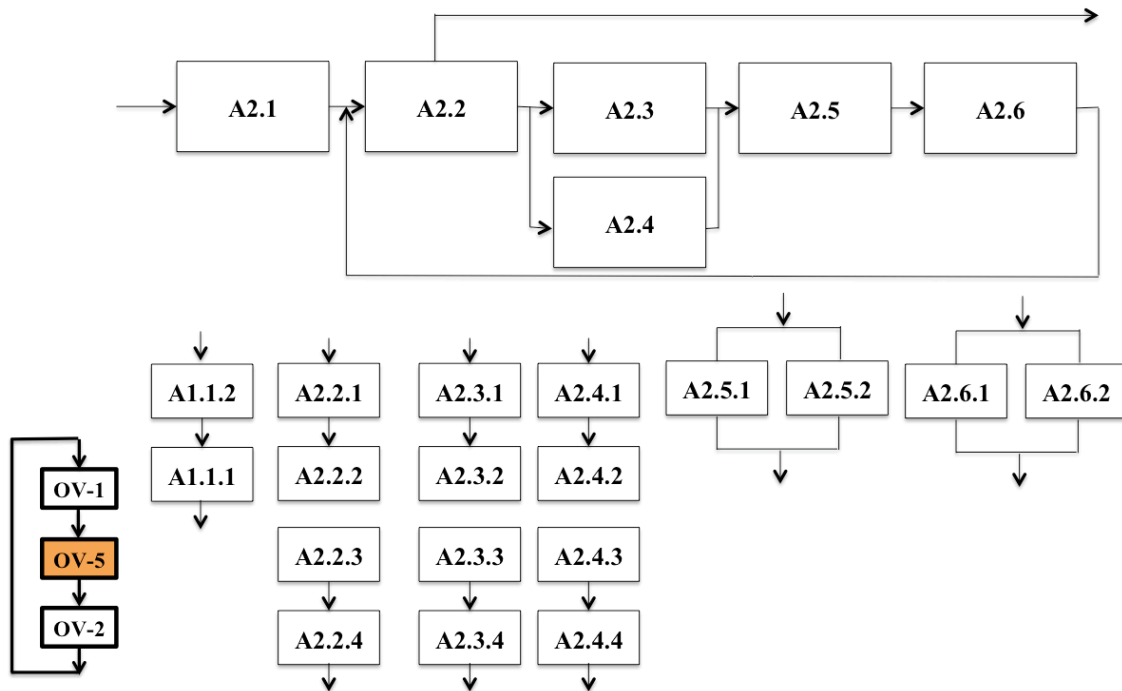


Figure 54. A2 operational activities flow of DHS of MicroDragon satellite

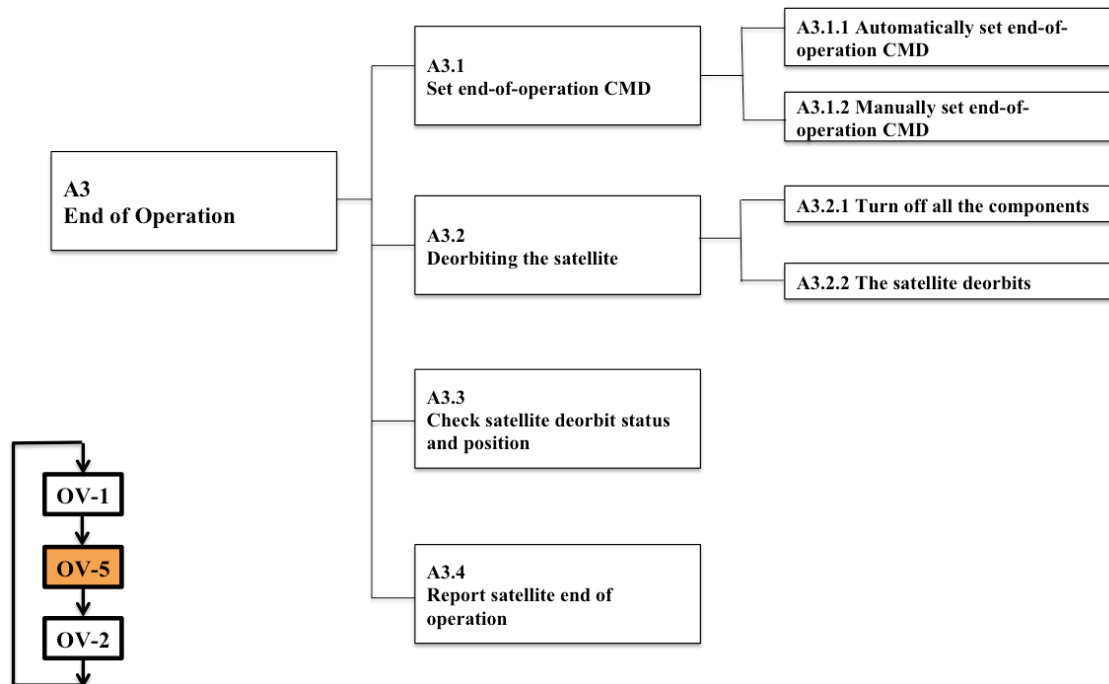


Figure 55. Operational activities in end-of-operations of DHS of MicroDragon (A3)

In end-of-operations phase, Figure 55 and Figure 56, shows broken down structure,

and flow of operational activities. MDG satellite has to implement deorbit by falling off to the ground. The deorbit command (A3.1) can be set automatically in such a way that the time of deorbit is set in the onboard software embedded in OBC (A3.1.1), or be set manually by uplink command from the ground (A3.1.2).

To prepare for deorbit (A3.2), it is necessary to check status and position of the satellite to guarantee that the satellite will fall off to a region where there is no or minimal damage to human and environment. Before deorbiting (A3.2.2), all the satellite components have to be turned off to avoid unexpected problems, which may happen during and after deorbiting (A3.2.1)

After successfully deorbit, identification of position of status of orbiting should be made to collect the debris (A3.3). Finally, the satellite owner should make a report/ announcement on orbiting status of the satellite.

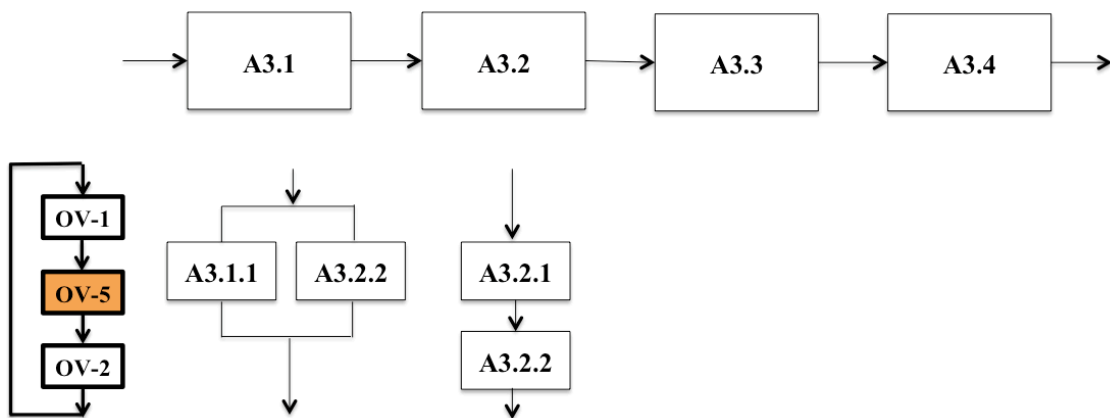


Figure 56. A3 operational activities flow of DHS of MicroDragon satellite

**OV-2** (Operational node connectivity): Operational node is an element of the operational architecture that produces, consumes, or processes information <sup>[24]</sup>. An operation node can represent an operational/ human role, an organization or organization type, etc.

DHS of MicroDragon satellite consists of three nodes including node A (MicroDragon satellite), node B (ground station), and node C (mission control center, MCC), as shown in Figure 57. Besides, there is an external source and destination, which is the end-users (or scientists) who needs the data for their researches.

Those nodes are connected via needlines. A needline documents the requirement to exchange information between nodes. Data going through a needline are also specified. Data from node A to node B, and from node B to node C contain telemetry and

command. Data from node B to node A, and from node C to node B include command. Node C communicates with external node by sending processed data and receiving back requirements, and feedbacks.

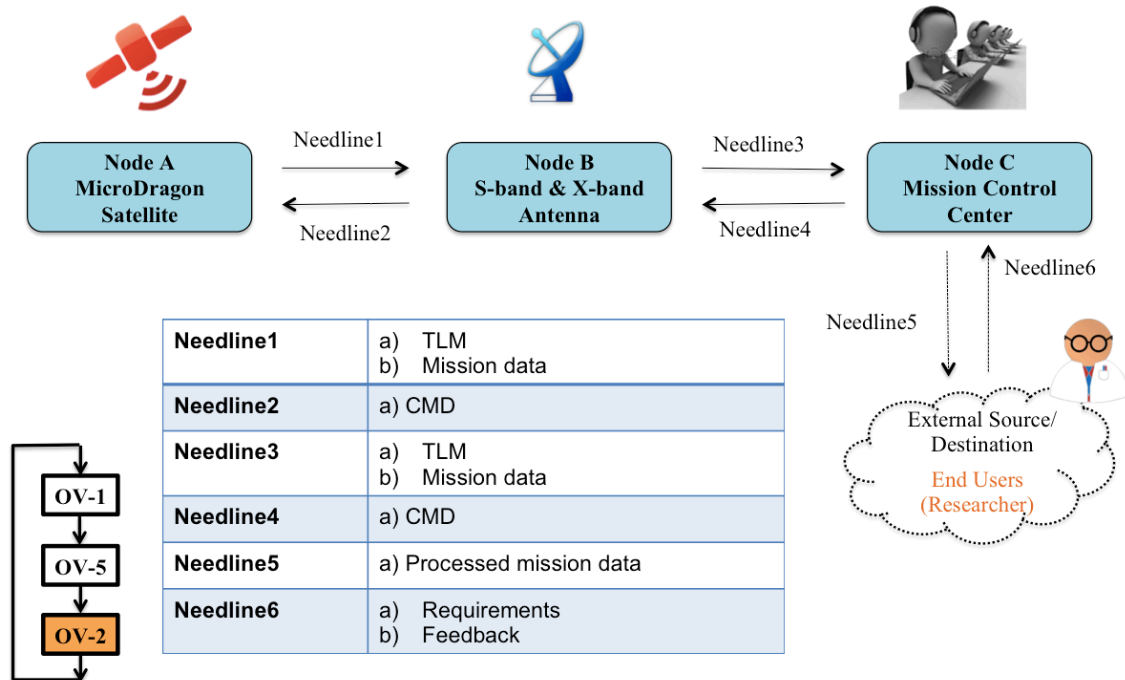


Figure 57. Operational nodes of DHS of MicroDragon satellite

Connection between OV-5 and OV-2 is shown in Figure 58 where operational activities are allocated to specific operational nodes, in other words, each operational node performs specific tasks.




| Node  | Activity Assignment   |
|---|---|
| <b>A</b><br> | <b>A1.2.1, A1.2.2, A1.3.1, A1.3.2, A1.3.7, A1.3.8, A1.4.1, A1.5.1, A2.2.3, A2.2.4, A2.3.1, A2.3.2, A2.4.1, A2.4.2, A2.4.3, A2.5.1, A2.6.1, A3.1.1, A3.2.2</b> |
| <b>B</b><br> | <b>A1.3.3, A1.3.6, A2.2.2, A2.3.3, A2.4.4</b>   |
| <b>C</b><br> | <b>A1.1.1, A1.1.2, A1.3.4, A1.3.5, A1.4.2, A1.5.2, A2.1.1, A2.1.2, A2.2.1, A2.3.4, A2.4.5, A2.5.2, A2.6.2, A3.1.2, A3.2.1, A3.3, A3.4</b>                     |

Figure 58. Operational activities assignment to operational nodes in DHS of MicroDragon satellite

As describes in Figure 58, the satellite (node A) perform operational activities including turning on satellite components right after separation (A1.2.1); supplying and

distributing power (A1.2.2); collecting, storing and packaging telemetry (A1.3.1); sending telemetry (A1.3.2); receiving CMD in initial phase (A1.3.7); processing and distributing CMD (A1.3.8); automatically detecting and analyzing failure in initial phase (A1.4.1); automatically recovery from failures in initial phase (A1.5.1); receiving CMD in nominal phase (A2.2.3); processing and distributing CMD in nominal phase (A2.2.4); collecting, storing and packaging TLM in nominal phase (A2.3.1); sending TLM in nominal phase (A2.3.2); capturing and recording mission data (A2.4.1); storing, processing and packaging mission data (A2.4.2); sending mission data (A2.4.3); automatically detecting and analyzing failures in nominal phase (A2.5.1); automatically recovery from failures (A2.6.1); automatically set end-of-operations CMD (A3.1.1); and the satellite deorbits (A3.2.2).

The ground station (node B) implement some operational activities including transmitting TLM in initial phase (A1.3.3); transmitting CMD in initial phase (A1.3.6); transmitting CMD in nominal phase (A2.2.2); transmitting TLM in nominal phase (A2.2.3); and transmitting mission data in nominal phase (A2.4.4).

The operational activities assigned to the mission control center (node C) are composed of starting satellite-tracking antenna system in initial phase (A1.1.1); starting up the mission control center in initial phase (A1.1.2); receiving, storing and analyzing TLM in initial phase (A1.3.4); generating CMD in initial phase (A1.3.5); manually detecting and analyzing failures in initial phase (A1.5.2); activating/ deactivating the satellite-tracking antenna system in nominal phase (A2.1.1); activating/ deactivating the mission control center in nominal phase (A2.1.2); generating CMD in nominal phase (A2.2.1); receiving, storing and analyzing TLM in nominal phase (A2.3.4); receiving, storing and processing mission data in nominal phase (A2.4.5); manually detecting and analyzing failures in nominal phase (A2.5.2); manually recovery from failures in nominal phase (A2.6.2); manually set end-of-operations CMD (A3.1.2); manually turning off all the satellite components (A3.2.1); checking satellite deorbiting status and position (A3.3); and reporting satellite deorbiting status (A3.4).

b) Systems view

**SV-4** (Systems functionality description): The systems functionality description documents system functional hierarchies and system functions, and the system data flows between them <sup>[24]</sup>. Applying system design approach to make functional design and physical design of DHS of MicroDragon satellite. In each phase of operations,

---

function requirements can be obtained by using context diagram and use case diagram. As mentioned in OV section, there are three phases of operations of DHS of MicroDragon satellite which are initial operations, nominal operations, and end-of-operations.

*In initial operations phase:* Context diagram defines the boundary between the systems, or parts of a system, and its environment, showing the entities that interact with it. DHS (System of Interest, SoI) consists of the satellite, X-band antenna, S-band antenna, and the mission control center. The SoI interacts with space environment as it receives space condition, such as temperature, solar radiation, micro-gravity effect, sunlight from the space. The operators lie outside of the boundary of the SoI, operate the satellite. The context diagram in initial phase is shown in Figure 59. In initial operations phase, only telemetry and command are transmitted via DHS.

The system of interest (DHS) consists of the satellite, the ground station and the mission control center. The satellite sends/receives telemetry/command to/from the mission control center, via the ground station.

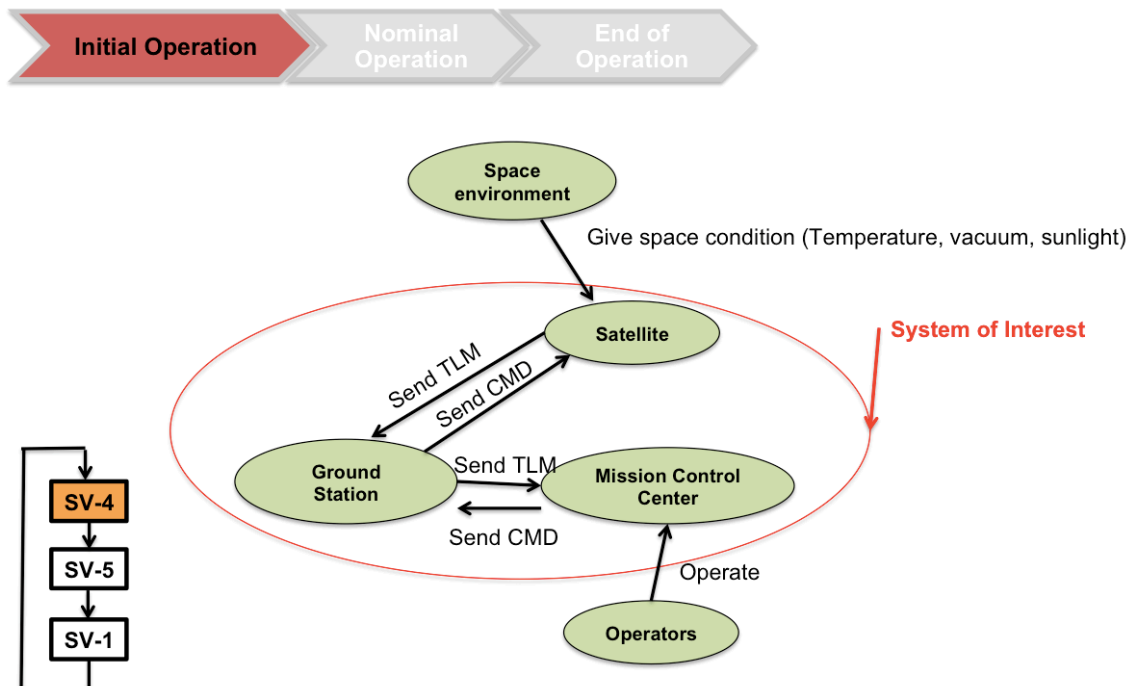


Figure 59. Context diagram of DHS of MicroDragon in initial operations phase

The use case diagram, shown in Figure 60, represents the user's interaction with the system. The diagram shows the relationship between the users and different use cases in which the user is involved.

Space environment provides heat for the satellite. The heat from the sun may cause temperature inside the satellite to increase beyond operational temperature ranges of satellite components, thus, damaging the components. Therefore, it is necessary to have temperature control inside the satellite to maintain temperature inside the satellite in allowable ranges. Besides, the sun also provides sunlight, which contains a big energy. The satellite can use it to maintain its operations in space by transforming solar energy into electric energy. In addition, the satellite must have ability to receive command and transmit telemetry from/ to the mission control center.

The ground station consisting of antennas system is used to transmit telemetry and command back and forth between the satellite and the mission control center, as in initial operations, there is no mission data.

In the mission control center (MCC), the operators are involved in generating and sending command, as well as receiving, storing and analyzing telemetry. The mission control center receive/ send TLM/ CMD from/to the satellite via the ground station.

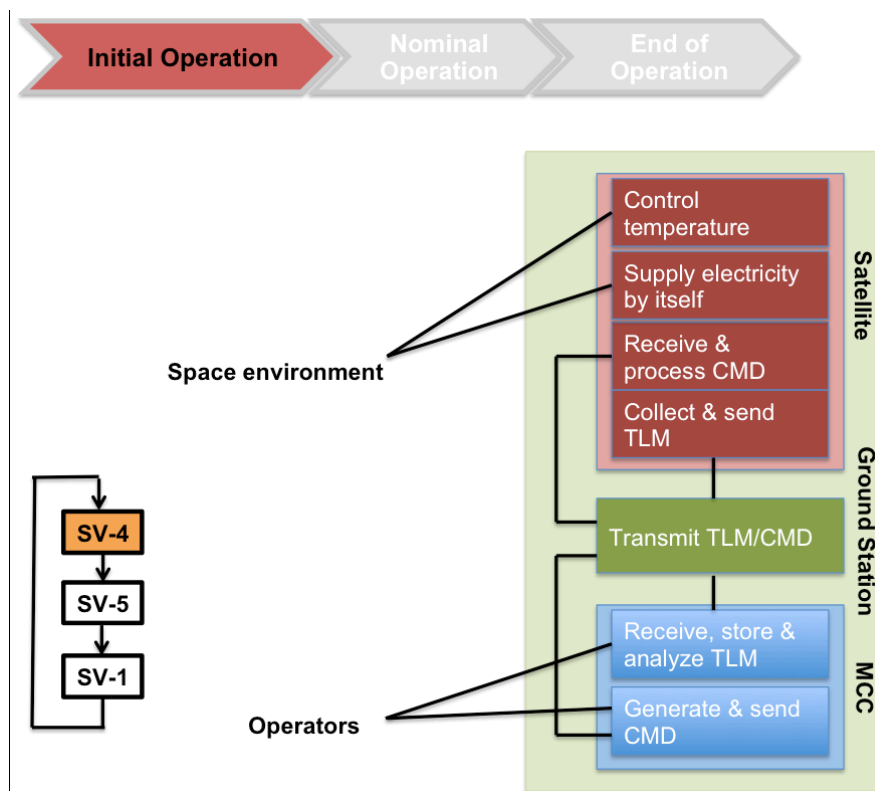


Figure 60. Use case diagram of DHS in initial operations phase of MicroDragon satellite

Functional requirements can be obtained from context diagram and use case diagram as in Figure 61.

Each operational node is assigned to specific high-level functions. The high-level

functions are then broken down to low-level ones in such a way that it is possible to allocate these low-level functions to one specific subsystem/ components inside each node.

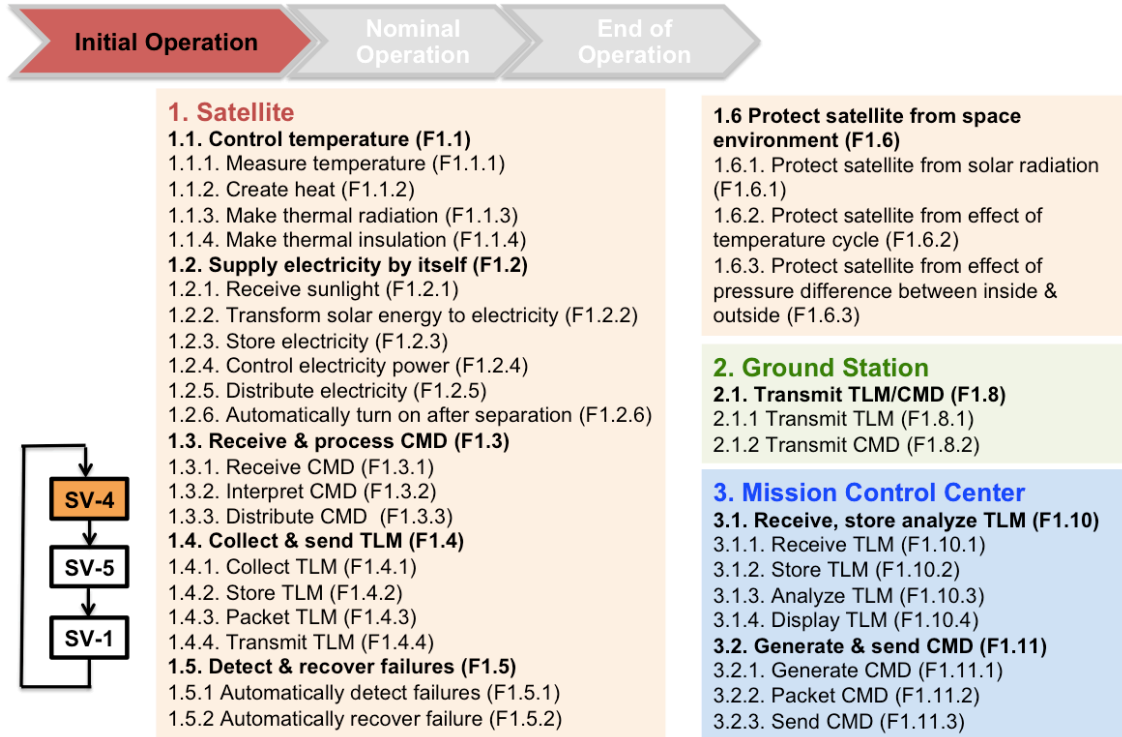


Figure 61. Functional requirements of DHS in initial operations phase of MicroDragon satellite

The satellite (node A), in initial phase, performs some functions including controlling temperature (F1.1), supplying power by itself (F1.2), receiving and processing CMD (F1.3), collecting and sending TLM (F1.4), detecting and recovering failures (F1.5), and protecting the satellite from space environment (F1.6). Control-temperature function (F1.1) can be broken down to low-level functions, such as, measuring temperature (F1.1.1), heating up function (F1.1.2), creating thermal radiation inside the satellite (F1.1.3), and creating thermal insulation from space environment (F1.1.4). To have the ability to supply power by itself (F1.2), the satellite should be able to receive sunlight (F1.2.1), transform solar energy to electric energy (F1.2.2), store electricity power (F1.2.3), control electricity power (F1.2.4), distribute electricity power to satellite components (F1.2.5), and automatically turn on power supply right after separation (F1.2.6). To receive and process CMD sent from the ground station (F1.3), the satellite has to have receiving CMD function (F1.3.1), interpreting CMD function (F1.3.2), and distributing CMD function (F1.3.3). For collecting and sending TLM (F1.4), some functions should be included in the satellite, such as, collecting TLM

functions (F1.4.1), storing TLM function (F1.4.2), packaging TLM function (F1.4.3), and transmitting TLM function (F1.4.4). The function of detection and recovery from failures (F1.5) can be broken down into automatically detecting failures function (F1.5.1), and automatically recovering from failures (F1.5.2). For protecting the satellite from space environment (F1.6), the satellite should contain some functions including protecting satellite from solar radiation (F1.6.1), protecting satellite from effect of temperature cycle that the temperature in space changes a lot in short time (for example from  $-100^{\circ}\text{C}$  to  $+100^{\circ}\text{C}$  in several hours) (F1.6.2), protecting satellite from effect of different pressure between its inside and outside (F1.6.3).

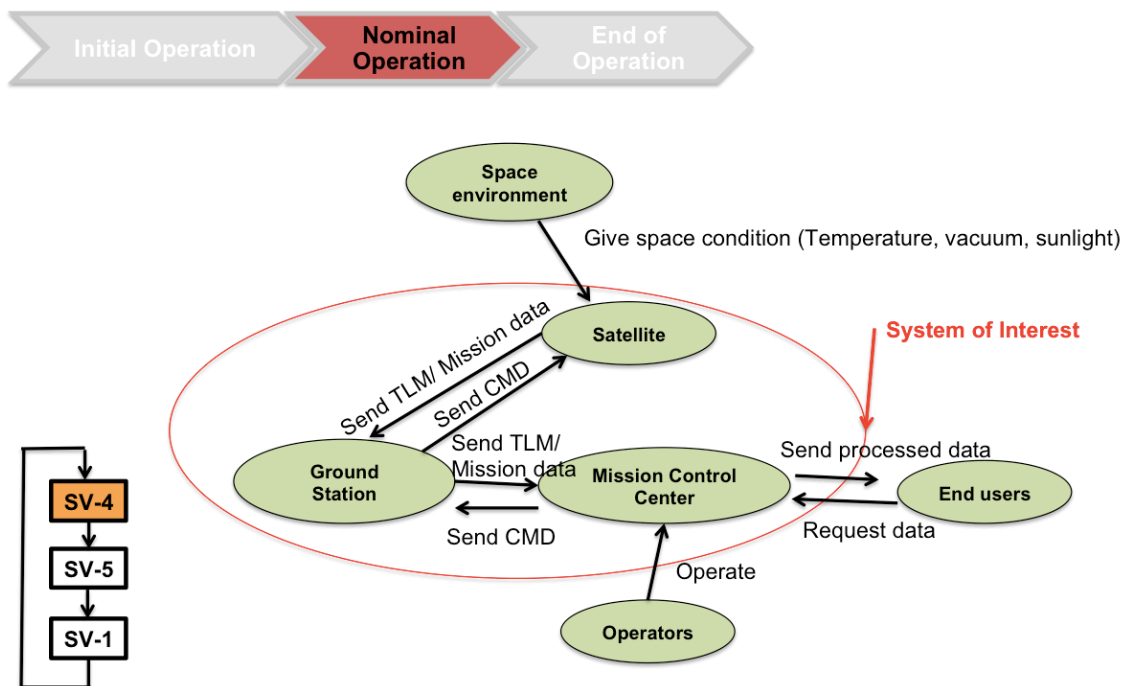


Figure 62. Context diagram of DHS in nominal operations phase of MicroDragon satellite

*In nominal operations phase:* The context diagram, Figure 62, shows interaction between the system of interest and the environment. The system of interest (DHS) gets space condition (temperature, vacuum, sun light, solar radiation, etc.) from space environment. Operators in the mission control center operate the satellite by sending CMD, and monitoring & analyzing TLM. In addition, the end-users interact with the system by sending feedback after receiving required data from the mission control center. In nominal phase, the satellite can perform the mission by capturing the pictures of targeted areas and implementing data downlink. This phase is different from the initial phase in a fact that both mission and housekeeping data are transmitted between the satellite and the ground segment in the nominal phase, while there is only

transmission of housekeeping data in the initial phase.

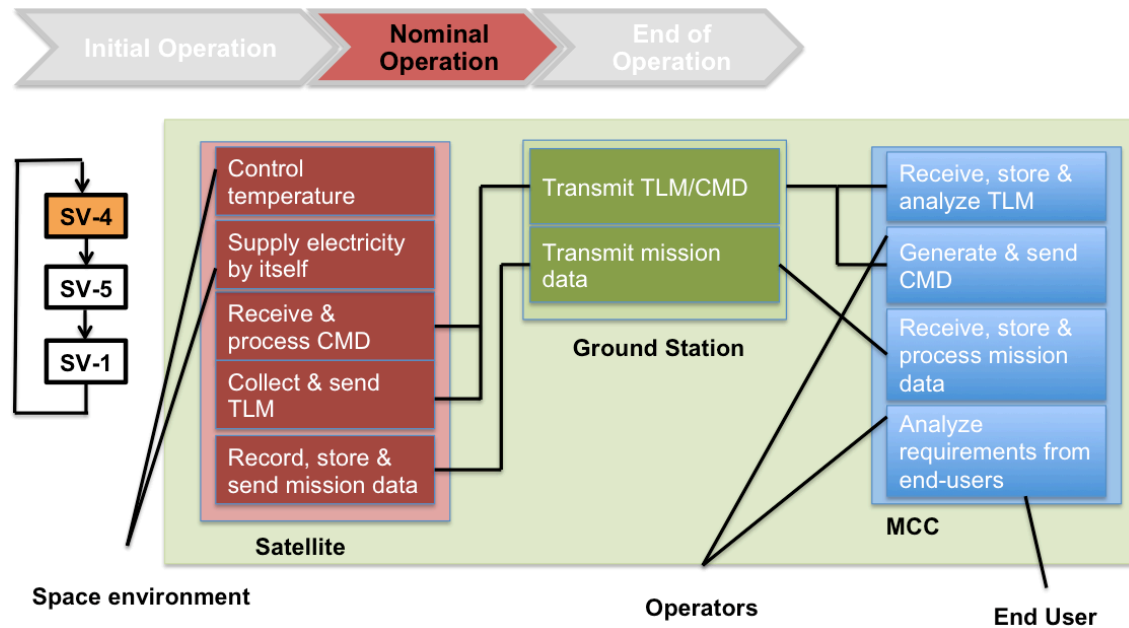


Figure 63. Use case diagram of DHS in nominal operations phase of MicroDragon satellite

The use case diagram in nominal phase, Figure 63, represents the interaction between the system of interest with the users including space environment, operators and end-users. In addition to the same use cases kept from the initial phase, some use cases related to mission data are considered, as in nominal phase, the satellite performs the mission. The first different use case comes from the satellite. The satellite has to record, store and process mission data from the payload devices in onboard memory. The second one comes from the mission control center, the mission control center receives, stores, and processes mission data. Besides, the mission control center has to analyze feedback and requirements from the end-users.

Functional requirements in nominal phase can be derived from the context diagram shown in Figure 62 and the use case diagram described in Figure 63. There are some functions similar to those in the initial phase. The differences lie on functions related to mission data.

The satellite has one high-level function to record, store, and send mission data (F2.7), as indicated in Figure 64. This high-level function can be broken down into several low-level functions including controlling satellite attitude over targeted areas (F2.7.1), recording required mission data (F2.7.2), storing mission data in onboard memory (F2.7.3), and sending mission data to the ground segment (F2.7.4). The

function F2.7.1 should be included in the satellite, as sometimes, the satellite cannot vertically above the targeted areas, thus, the satellite attitude should be controlled to point to the want-to-capture-pictures areas.

In addition to the same function with the initial phase, in the nominal phase, the ground station has one more function, which is transmission of mission data (F2.9). Figure 65 shows the functional requirements. When the satellite can communicate with the ground station, the collected mission data should be sent to the mission control center in order to store, process, and provide for the end-users.

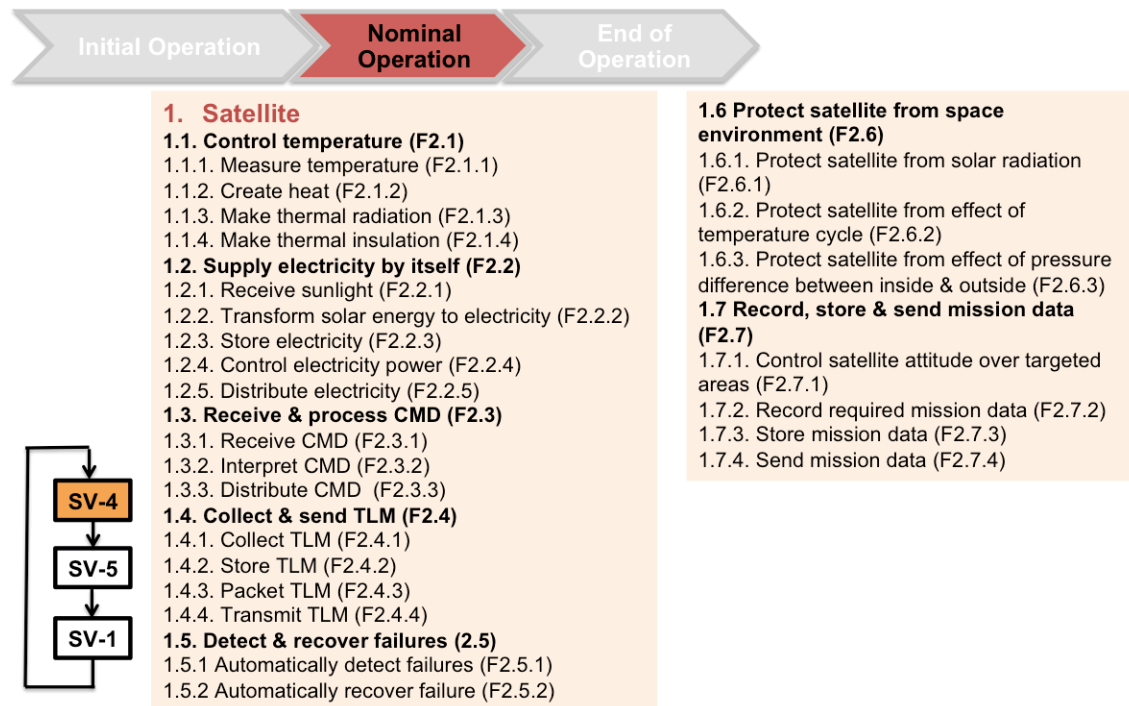


Figure 64. Functional requirements (1) of DHS in nominal operations phase of MicroDragon satellite

The functional requirements of mission control center in nominal phase are added to those in initial phase by two high-level functions. The first one is receiving, storing and processing mission data function (F2.12). This function is composed of several low-level functions including receiving mission data (F2.12.1), storing mission data (F2.12.2), and processing mission data (F2.12.3).

The second one is analyzing requirements from the end-users (F2.13). This function is used to identify what kind of data the end-users want to have, which areas the satellite has to take the pictures, etc. This function also contains some low-level functions including getting requirements from end-users (F2.13.1), interpreting requirements (F2.13.2), and planning mission operations (F2.13.3).

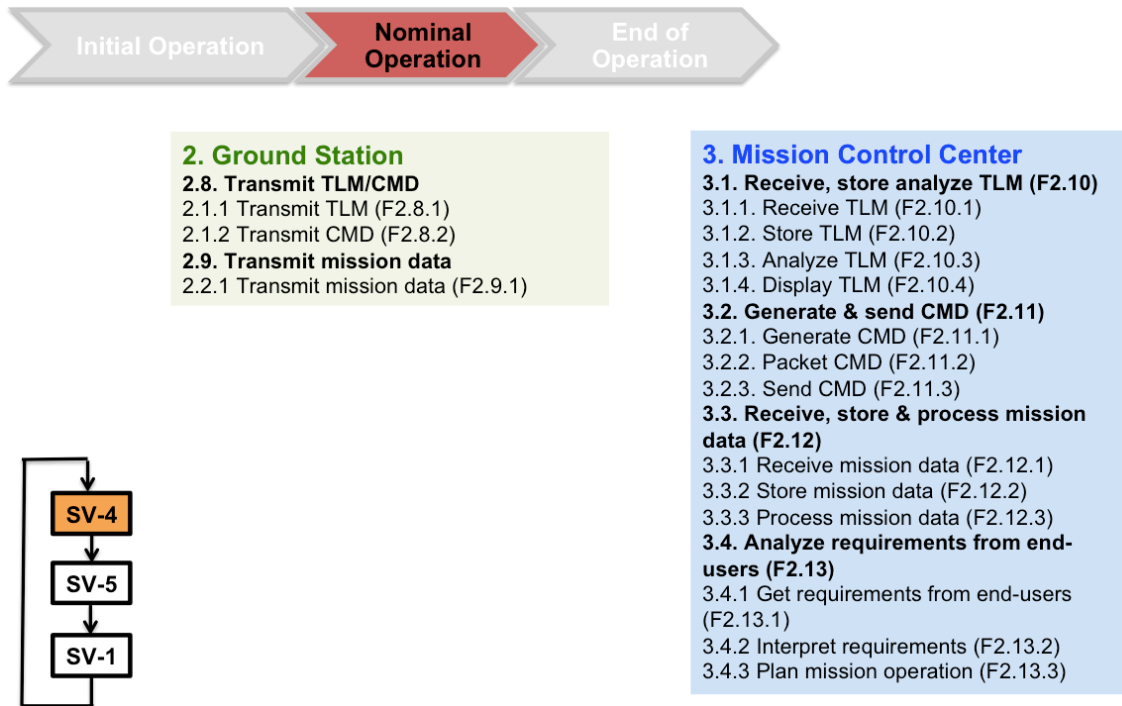


Figure 65. Functional requirements (2) of DHS in nominal operations phase of MicroDragon satellite

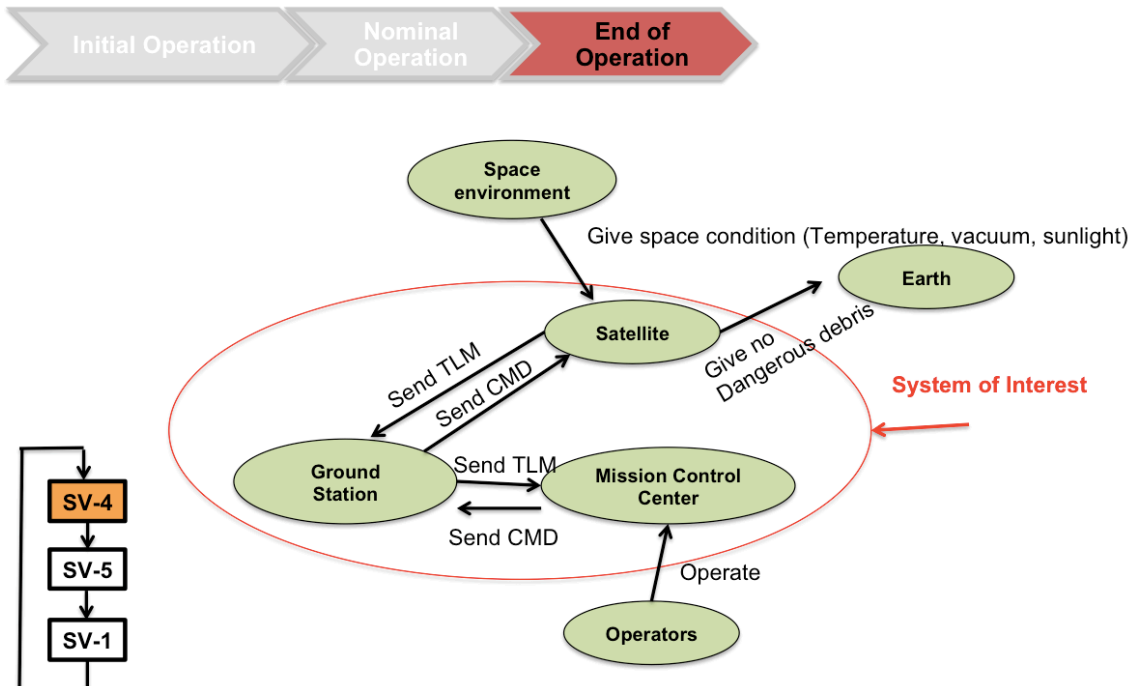


Figure 66. Context diagram of DHS in end-of-operations phase of MicroDragon satellite

*In end-of-operations phase:* in end-of-operations phase, the satellite has falls off to the earth and it must guarantee that the debris from the satellite gives no harm and damage to people and environment, as it is shown in Figure 66. As the satellite does not do the mission anymore, there is no mission data going through the system of interest

(DHS), only TLM/ CMD are shared between the satellite and the ground segment. The operator is necessary to control the satellite deorbit.

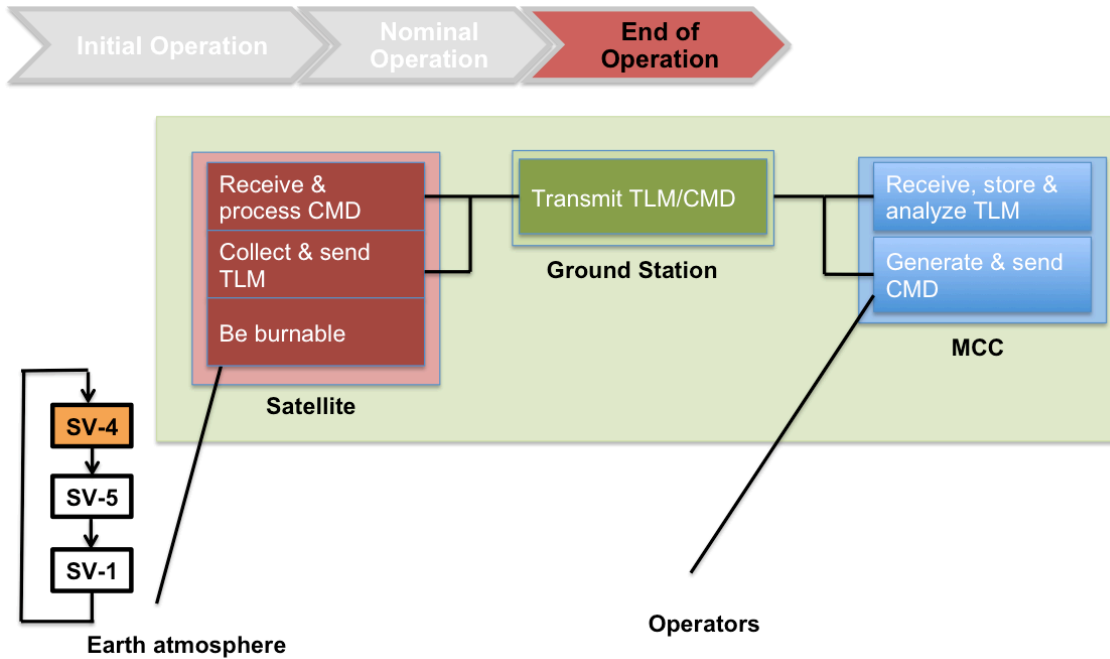


Figure 67. Use case diagram of DHS in end-of-operations phase of MicroDragon satellite

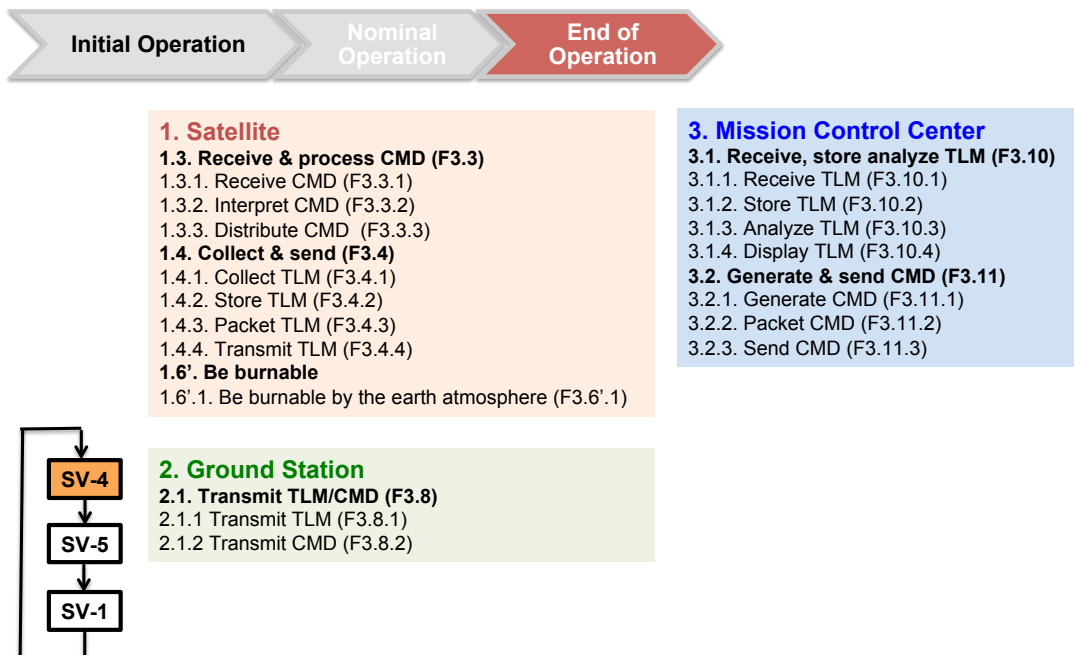


Figure 68. Functional requirements of DHS in end-of-operations phase of MicroDragon satellite

In Figure 67, the use case diagram, when the satellite falls off to the ground through the earth atmosphere, the friction resulted by collision of air molecules on surfaces of the satellite, causes the heat to burn the satellite. To guarantee that, there is no or

minimal harm and damage to human and environment, the satellite should be made of burnable materials.

Functional requirements derived from the context diagram and use case diagram in end-of-operations phase are shown in Figure 68. One new function is added in this phase, is that the feature of the satellite materials should include burnable property (F1.6'). While all functions of the ground station and the mission control center are the same with those in initial phase.

To describe sequences of functions implemented in satellite data handling system, function flow block diagram (FFBD) is used, as shown in Figure 69. The satellite starts from initial operations to nominal operations, and ends at end-of-operations. In each high-level function, the sequences of low-level functions are defined.

For example, some sequences including sequence of temperature control, sequence of supplying electric power, sequence of transmission of TLM/ CMD, and sequence of performing a mission, are to be described as bellows.

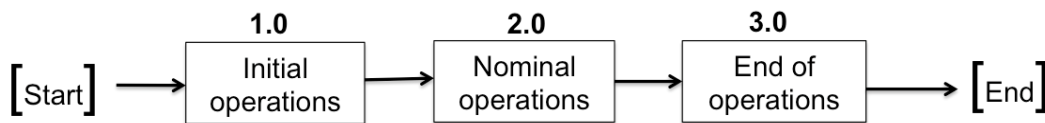


Figure 69. Function flow block diagram of DHS of MicroDragon satellite

Figure 70 shows the sequence of temperature control in initial operations phase, the temperature of the satellite will be observed by temperature measuring function (F1.1.1), and controlled by thermal radiation function (F1.1.3), thermal insulation function (F1.1.4), and heating up function (F1.1.2).

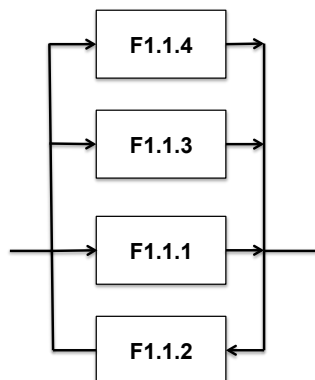


Figure 70. Sequence of temperature control in initial operations phase of MicroDragon satellite

Figure 71 describes the sequence of supplying electric power in nominal operations. The satellite receives solar energy from the sunlight (F2.2.1), and transforms it to

electric energy (F2.2.2). The electric power should be appropriately controlled (F2.2.4), and distributed to relevant satellite components (F2.2.5). As the satellite cannot receive sun light in eclipse, it should have enough amount of electric power stored during sunshine to maintain the operations (F2.2.3).

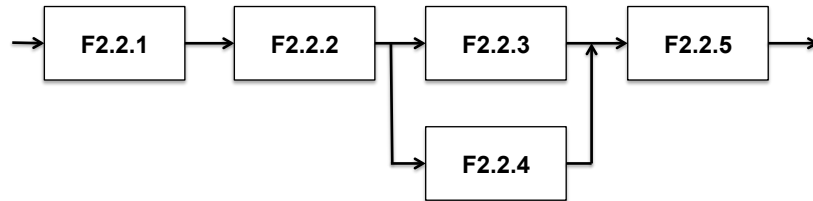


Figure 71. Sequence of supplying electric power in nominal phase of MicroDragon satellite

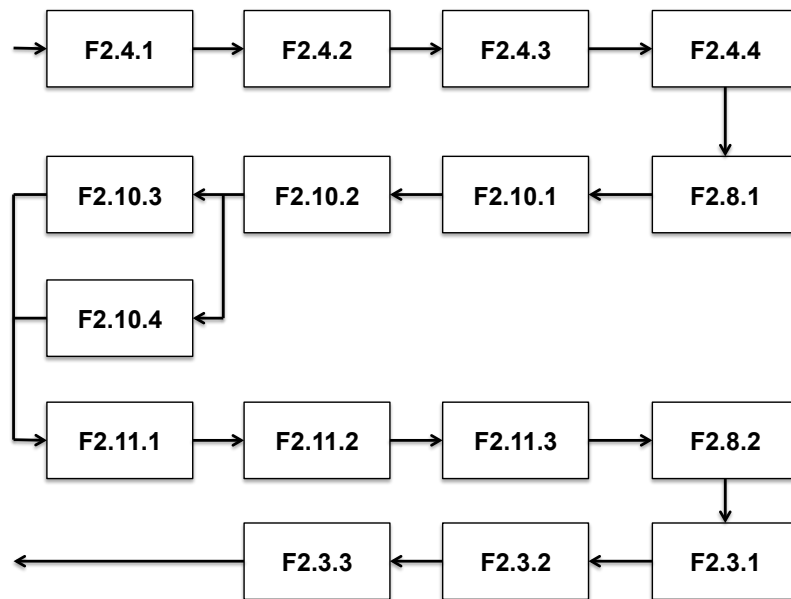


Figure 72. Sequence of TLM/ CMD transmission in nominal phase of MicroDragon satellite

Figure 72 shows the sequence of transmission of TLM/CMD in nominal phase. The satellite collects telemetry from satellite components (F2.4.1) → store the TLM in onboard memory (F2.4.2) → packaging TLM in TLM frames (F2.4.3) → sending TLM via satellite antenna (F2.4.4) → the telemetry is transmitted via ground station antenna to the mission control center (F2.8.1) → the mission control center receives the TLM (F2.10.1) → the mission control center stores the TLM (F2.10.2), analyzes the TLM (F2.10.3) and displays the TLM (F2.10.4) → the mission control center generates CMD based on TLM analysis results (F2.11.1) → packaging CMD (F2.11.2) → sending CMD via the ground station antenna (F2.11.3) → the ground station antenna transmits the CMD package (F2.8.2) → the satellite receives the CMD (F2.3.1) → the satellite interprets the CMD (F2.3.2) → the satellite onboard computer distributes CMD to

relevant components (F2.3.3).

Figure 73 describes the sequence of performing mission in nominal phase. The mission control center generates CMD to take pictures (F2.11.1) → the CMD is packaged (F2.11.2) and sent via the ground station antenna (F2.11.3) → the ground station antenna transmits the CMD to the satellite (F2.8.2) → the satellite control the attitude pointing to the targeted area (F2.7.1) → the camera captures the pictures (F2.7.2) and stores in onboard memory (F2.7.3) → the satellite sends the mission data via the satellite antenna (F2.7.4) → the ground station transmits mission data to the mission control center (F2.9.1) → the mission control center receives mission data (F2.12.1) → the mission control center processes mission data (F2.12.3) and stored it (F2.12.2).

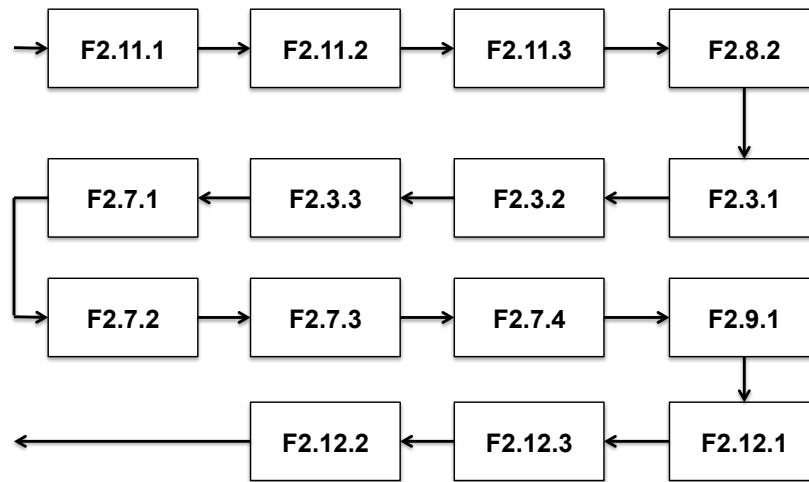


Figure 73. Sequence of performing mission in nominal phase of MicroDragon satellite

**Physical design of DHS of MicroDragon satellite** is shown in Figure 74. The satellite is composed of some subsystems including command & data handling subsystem to control and manage all the tasks, processes in the satellite; attitude determination and control subsystem to control the attitude of the satellite; electrical power supply subsystem to supply electric power to satellite components. Thermal subsystem to control and monitor temperature inside the satellite within allowable ranges; Communication subsystem to transmit housekeeping data and mission data between the satellite and the ground segment; mission subsystem to perform the mission (capturing pictures); and finally, structure subsystem to keep the satellite in shape, and protect the satellite from space environment and other effects from the rockets during launching.

The ground station includes S-band antenna to transmit TLM and CMD, X-band

antenna to transmit mission data between the satellite and the mission control center.

The mission control center includes some subsystems, such as, operation control& monitor for operating the satellite, CCSDS transport service for coding and decoding data, and data processing of raw data.

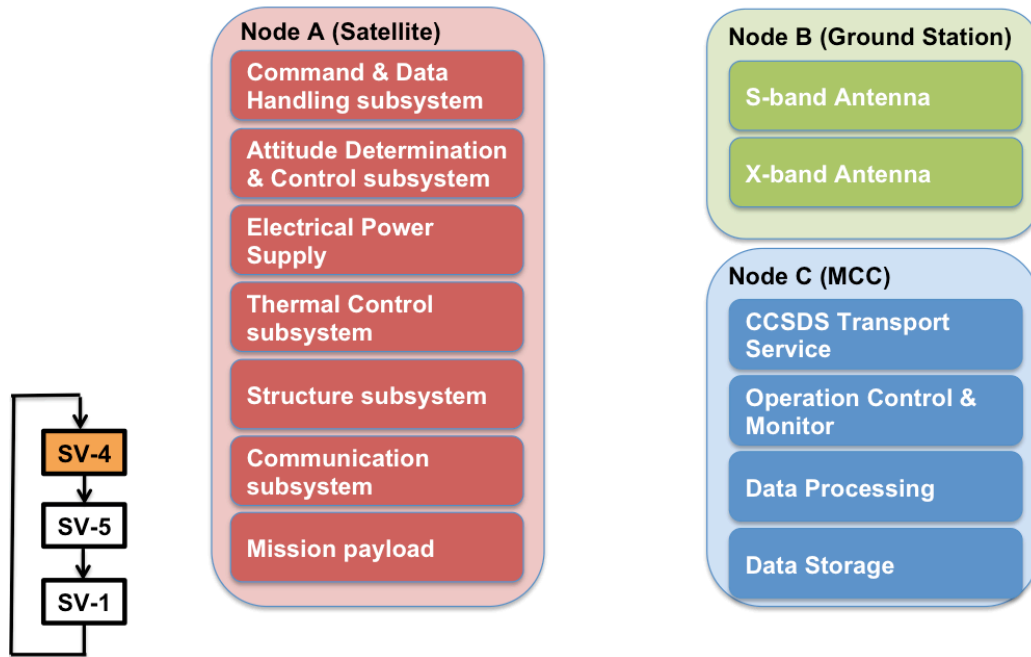


Figure 74. Physical design of DHS of MicroDragon satellite

**Functional allocation:** Based on physical design of DHS, all the functions can be allocated to relevant subsystems as shown in Table 8.

Table 8. Functional allocation of DHS of MicroDragon satellite

| Physical subsystems                          | Functions  |
|--|--|
| <i>Satellite</i>                             |  |
| Command & data handling subsystem            | F1.3.2, F1.3.3, F1.4.1, F1.4.2, F1.4.3, F1.5.1, F1.5.2, F2.3.2, F2.3.3, F2.4.1, F2.4.2, F2.4.3, F2.5.1, F2.5.2, F3.3.2, F3.3.3, F3.4.1, F3.4.2, F3.4.3 |
| Attitude determination and control subsystem | F2.7.1   |
| Thermal subsystem                            | F1.1.1, F1.1.2, F1.1.3, F1.1.4, F2.1.1, F2.1.2, F2.1.3, F2.1.4   |
| Structure subsystem                          | F1.6.1, F1.6.2, F1.6.3, F2.6.1, F2.6.2, F2.6.3, F1.6'.1  |
| Electrical power supply subsystem            | F1.2.1, F1.2.2, F1.2.3, F1.2.4, F1.2.5, F1.2.6, F2.2.1, F2.2.2, F2.2.3, F2.2.4, F2.2.5   |
| Communication                                | F1.4.4, F1.3.1, F2.3.1, F2.4.4, F3.3.1, F3.4.4, F2.7.4   |

|   |  |
|---|--|
| subsystem                                   |  |
| Mission subsystem                           | F2.7.2, F2.7.3   |
| <b><i>Ground station</i></b>                |  |
| S-band antenna                              | F1.8.1, F1.8.2, F2.8.1, F2.8.2, F3.8.1, F3.8.2   |
| X-band antenna                              | F2.9.1   |
| <b><i>Mission control center</i></b>        |  |
| Satellite operations<br>monitor and control | F1.10.3, F1.10.4, F1.11.1, F2.10.3, F2.10.4, F2.11.1,<br>F2.12.2, F1.12.3, F2.13.1, F2.13.2, F1.13.3 |
| CCSDS transport<br>service                  | F1.10.1, F1.11.2, F1.11.3, F2.10.1, F2.11.2, F2.11.3,<br>F2.12.1                                     |
| Ground data storage                         | F1.10.2, F2.10.2,  |

**SV-5** (Operational activity to system function traceability matrix): Defines the mapping between operational activities and systems functionality as shown in Table 9.

**Table 9. Mapping table between operational activities and systems functions**

| <b>Activities</b> | <b>Functions</b>  |
|-------------------|---|
| A1.1.1            | -   |
| A1.1.2            | -   |
| A1.2.1            | F1.2.6  |
| A1.2.2            | F1.2.5  |
| A1.3.1            | F1.4.1, F1.4.2, F1.4.3  |
| A1.3.2            | F1.4.4  |
| A1.3.3            | F1.8.1  |
| A1.3.4            | F1.10.1, F1.10.2, F1.10.3, F1.10.4                                |
| A1.3.5            | F1.11.1, F1.11.2  |
| A1.3.6            | F1.8.2  |
| A1.3.7            | F1.3.1  |
| A1.3.8            | F1.3.2, F1.3.3  |
| A1.4.1            | F1.5.1  |
| A1.4.2            | F1.4.1, F1.4.2, F1.4.3, F1.4.4, F1.8.1, F1.10.1, F1.10.3, F1.10.4 |
| A1.5.1            | F1.5.2  |
| A1.5.2            | F1.11.1, F1.11.2, F1.11.3, F1.8.2, F1.3.1, F1.3.2, F1.3.3         |

---



---

|        |   |
|--------|---|
| A2.1.1 | -   |
| A2.1.2 | -   |
| A2.2.1 | F2.11.1   |
| A2.2.2 | F2.8.2  |
| A2.2.3 | F2.3.1  |
| A2.2.4 | F2.3.2, F2.3.3  |
| A2.3.1 | F2.4.1, F2.4.2, F2.4.3  |
| A2.3.2 | F2.4.4  |
| A2.3.3 | F2.8.1  |
| A2.3.4 | F2.10.1, F2.10.2, F2.10.3, F2.10.4                                |
| A2.4.1 | F1.7.1, F1.7.2  |
| A2.4.2 | F2.7.3  |
| A2.4.3 | F2.7.4  |
| A2.4.4 | F2.9.1  |
| A2.4.5 | F2.12.1, F2.12.2, F2.12.3   |
| A2.5.1 | F2.5.1  |
| A2.5.2 | F2.4.1, F2.4.2, F2.4.3, F2.4.4, F2.8.1, F2.10.1, F2.10.3, F2.10.4 |
| A2.6.1 | F2.5.2  |
| A2.6.2 | F2.11.1, F2.11.2, F2.11.3, F2.8.2, F2.3.1, F2.3.2, F2.3.3         |
| A3.1.1 | -   |
| A3.1.2 | F3.11.1, F3.11.2, F3.11.3, F3.8.2, F3.3.1, F3.3.2, F3.3.3         |
| A3.2.1 | F3.11.1, F3.11.2, F3.11.3, F3.8.2, F3.3.1, F3.3.2, F3.3.3         |
| A3.2.2 | -   |
| A3.3   | -   |
| A3.4   | -   |

---

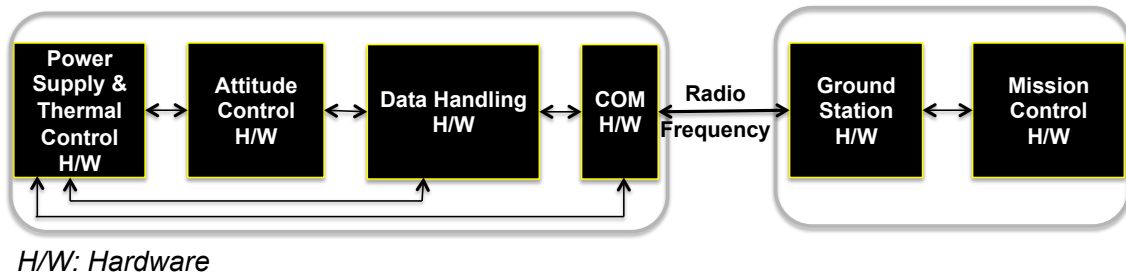


---

**SV-1** (Interface between system): Based on operational, physical and functional design of architecture of DHS, horizontal architecture can be derived, as shown in Figure 75.

The satellite consists of 2 onboard computers, which are OBC and science data handling unit (SHU). The OBC is embedded with 2 onboard software programs including ADCS software and C&DH software. The ADCS software is used to collect

telemetry from ADCS sensors, and help the C&DH software to give decisions by appropriate algorithms. The C&DH software decides to send appropriate CMD to ADCS actuators to control the attitude. All the TLM/ CMD coming from/ going to the satellite have to pass CCSDS transport service embedded in OBC. The CCSDS transport service is connected to S-band transponder (STRX) of communication subsystem.



**Figure 75. Horizontal architecture of DHS of MicroDragon satellite**

In science data handling unit, mission data processing software has interface with mission payloads and CCSDS transport service embedded in SHU. This CCSDS transport service is connected to X-band transmitter (XTX).

Radio frequency link between the satellite and the ground segment is described as the frequency link between the satellite antennas and the ground station antennas.

In mission control center, there is also a CCSDS transport service to code and decode the data sent from/ received in the mission control center. In addition, the mission control center also includes operations monitor and control, which is used to generate command, receive and analyze telemetry.

All the interfaces and links in DHS of MicroDragon are shown in Figure 75.

#### *4.2.1.2. Vertical architecture design of DHS of MicroDragon satellite*

Vertical architecture designed in this research adds upper-layers data interfaces to those, which are already described in horizontal architecture. The upper-layers data interfaces in vertical architecture are categorized into 3 layers including application layer, data link layer and physical link layer. The way of design of vertical architecture allows verification of upper-layers data interfaces at low-level test, and results in shortened unexpected iteration development processes, thus, reducing development risks of increasing development cost and time compared to that verification of those upper-layers interfaces is postponed until system-level tests.

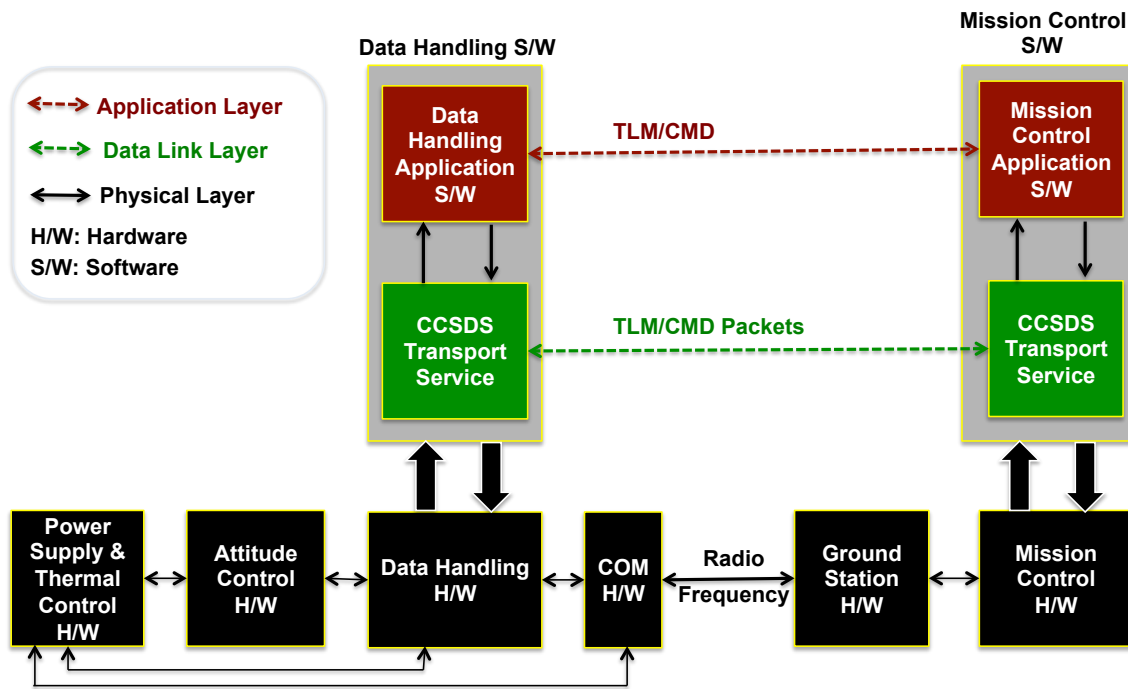


Figure 76. Vertical architecture of DHS of MicroDragon satellite

As it is described in Figure 76, three layers of interfaces of vertical architecture exist in DHS of MicroDragon satellite. The application interfaces consists of ADCS TLM interface between ADCS software embed in OBC and operations monitor and control of the mission control center, mission data interface between mission data processing software and operations monitor and control, and TLM/CMD interface between C&DH software and operations monitor and control. The data link interface connects two CCSDS transport services of OBC and SHU to CCSDS transport service of the mission control center. The RF interface describes the connection between the satellite antennas and the ground station antennas.

#### 4.2.2. Verification of MicroDragon DHS

Verification of MicroDragon DHS is implemented by using TableSat facilities in Tokyo University. Because of lacking mission data and mission payload components, verification of mission data cannot be done. The TableSat layout is in Figure 17.

To test TLM/CMD interface and physical link interface, the testing layout is set up as in Figure 77. This figure also shows data interfaces between DAS computer and PCU/ PCDU in application layer (red-colored lines), and between DAS computer and MOBC in data link layer (green-colors line).

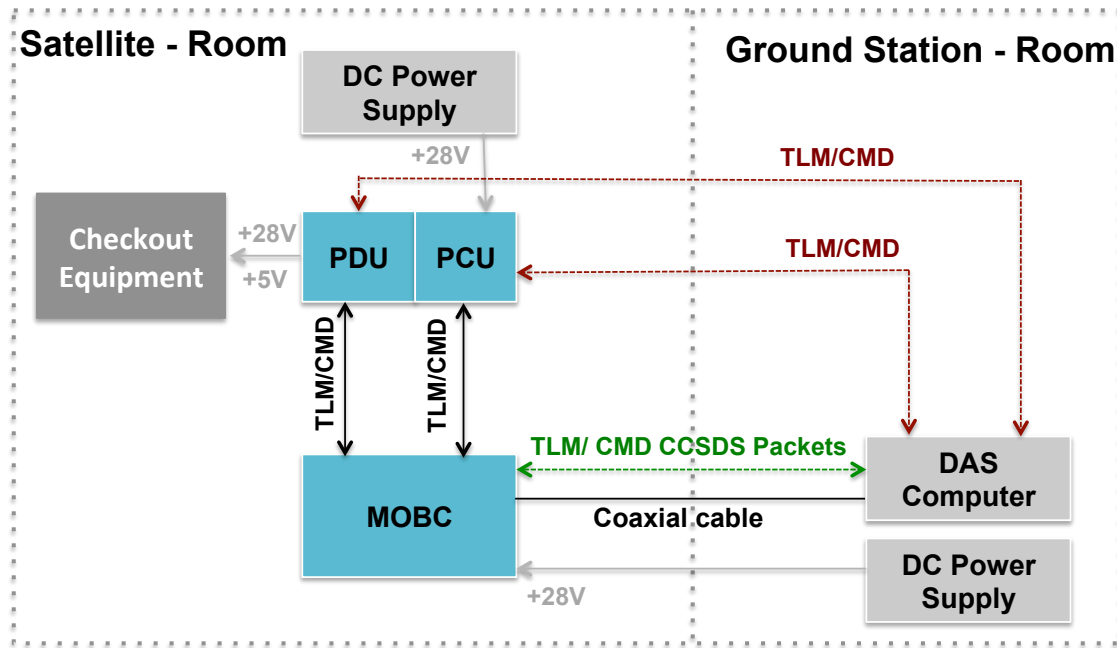


Figure 77. Testing TLM/ CMD layout with TableSat in Tokyo University

The testing configuration is set as in Figure 80. DAS computer is connected to onboard computer (MOBC) via cables. MOBC receives TLM from and send CMD to PCU/PDU via RS422 interface. PCU and PDU have internal connection. PCU is supplied by a +28V-direct current (DC) power. PDU supplies +5VDC power to checkout equipment, which plays as satellite components (for example, gyro, sun sensor, etc.).

Verification of CMD and TLM: Sending CMD from DAS computer to MOBC to turn on/ off satellite components (check out equipment), illustrated in Figure 78. If one the component is on, the corresponding LED of checkout component will be on. Therefore, it can check ON/ OFF status of components by checking LEDs status in checkout component. There is a camera put in ground station room to observe LEDs on/off status, as it is shown in Figure 79.

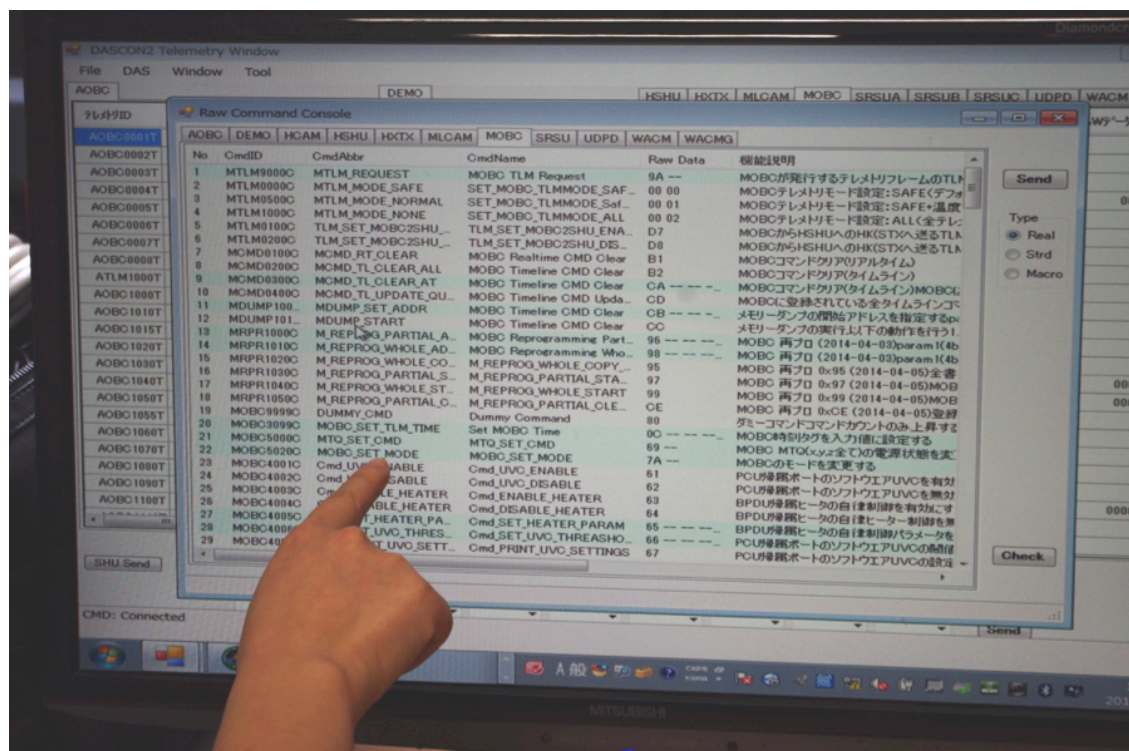


Figure 78. Sending CMD from DAS computer to MOBC in one TableSat activity

Reception of TLM in ground segment is confirmed by checking telemetry monitor screen, illustrated in Figure 81.

Description of experiment: The experiment procedures are described as in Table 10. The experiment was implemented through multiple steps. At first, DAS computer and MOBC were turned on before starting sending CMD from DAS to MOBC. In other parts, PCU was turned on first because it supplied electrical power to PDU, and PDU supplied electrical power for other components. In this experiment, PCU and PDU supplied electrical power to turn on LEDs in checkout equipment. After sending CMD, some TLM related to the CMD, which showed that MOBC had already received CMD, such as CMD counter in MOBC, were checked. Besides, some other TLM were also monitored as evidence of implementation of the CMD.

Table 10. Experiment procedures in one TableSat activity

| Step | Actions  |
|------|--|
| 1    | Turn on DAS computer                                     |
| 2    | Turn on MOBC by turning on +28VDC supply of MOBC         |
| 3    | Send CMD to turn on PCU                                  |
| 4    | Check TLM which confirms implementation of CMD at step 3 |

---



---

|     |   |
|-----|---|
| 5   | Check voltage and current TLM of PCU in order to confirm that PCU is on                                 |
| 6   | Send CMD to turn on PDU   |
| 7   | Check TLM which confirms implementation of CMD at step 6  |
| 8   | Check voltage and current TLM of PDU in order to confirm that PDU is on                                 |
| 9   | Send CMD to turn on gyro  |
| 10  | Check TLM which confirms implementation of CMD at step 9  |
| 11  | Confirm ON status of gyro by checking on/off status of LED corresponding to gyro in checkout equipment  |
| 12  | Send CMD to turn off gyro   |
| 13  | Check TLM which confirms implementation of CMD at step 12   |
| 14  | Confirm OFF status of gyro by checking on/off status of LED corresponding to gyro in checkout equipment |
| ... |   |
| ... | Similar steps were applied to turn on/off other satellite components                                    |
| ... |   |
| N   | Send CMD to turn off PDU  |
| N+1 | Check TLM which confirms implementation of CMD at step N  |
| N+2 | Check voltage and current TLM of PDU in order to confirm that PDU is off                                |
| N+3 | Send CMD to turn off PCU  |
| N+4 | Check TLM which confirms implementation of CMD at step N+3  |
| N+5 | Check voltage and current TLM of PCU in order to confirm that PCU is off                                |
| N+6 | Turn off MOBC by turning off +28VDC supply of MOBC  |
| N+7 | Turn off DAS computer   |

Experiment results: All the results obtained by checking monitor screen and the camera confirmed that all the commands sent from DAS computers had been implemented properly, and showed that telemetry responses were as expected.



Figure 79. Checking LEDs status in checkout equipment in one TableSat activity

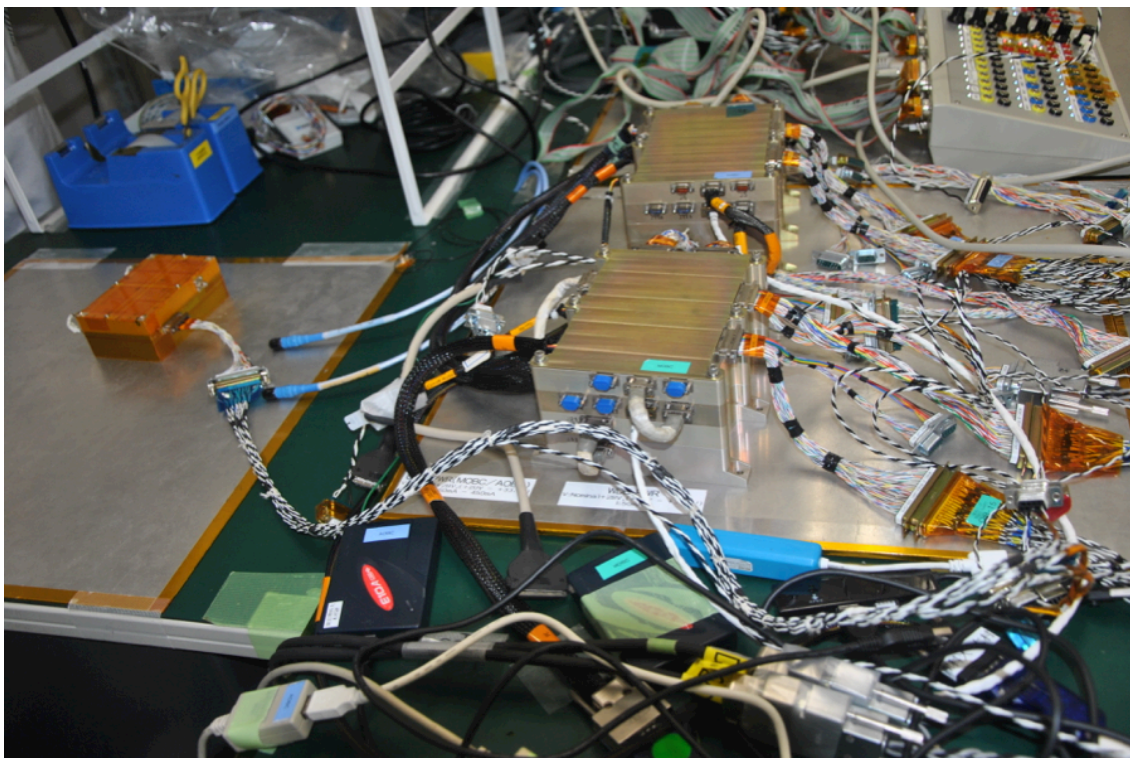


Figure 80. Connecting components with cables in one TableSat activity

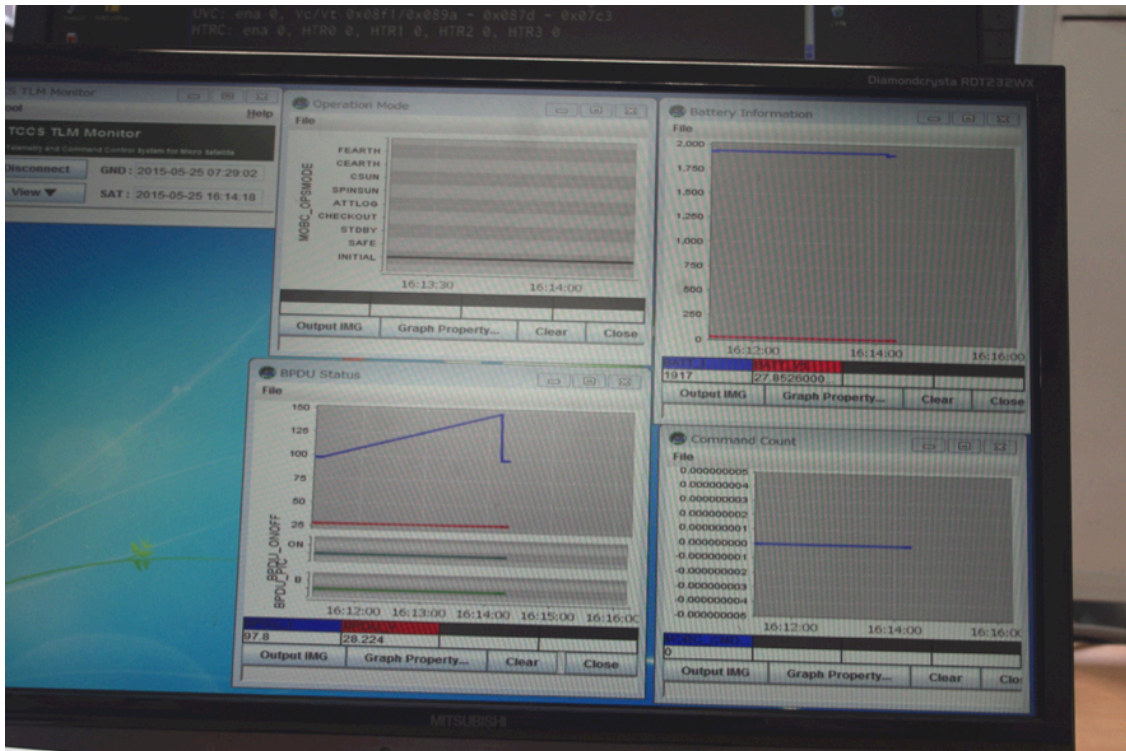


Figure 81. Monitoring TLM in one TableSat activity

#### 4.3. Applying the approach to develop DHS of the data relay satellite system

The data relay satellite system is designed to increase the amount of time that the satellite is in communication with the ground station, and improve the amount of data, which can be transferred.

In this research, an imaginary data relay satellite system is introduced, consisting of a low-earth-orbit (LEO) satellite and its ground segment, and a geostationary (GEO) satellite and its ground segment. The data relay satellite system is shown in Figure 47. The mechanism of the data relay satellite system is as follows. The LEO satellite sends its collected data to the GEO satellite, then the GEO satellite downlink the data to its GEO ground segment. Followed by that the GEO ground station sends the data to the LEO ground station by dedicated cables, such as Sinet in Japan. As the GEO satellite can always communicate with its ground station and the amount of time that the LEO satellite can talk with the GEO satellite is bigger than the time amount that the LEO satellite can talk with its ground station, thus, increasing the amount of time in communication between the LEO satellite and its ground station.

The approach is applied to design and verify data handling system of the data relay satellite system. DHS design of the data relay satellite system includes horizontal

architecture, and vertical architecture.

The horizontal architecture is illustrated in Figure 82. Mission data and housekeeping data of the satellite 1 are collected by science data handling unit and onboard computer, and then these data are packed and sent to communication subsystem of the satellite 1 by CCSDS transport service 1. The satellite 2 can receive these data by antenna systems in two satellites via RF layer. These data are stored in onboard memory of the satellite 2, and then sent to the ground segment 2. From the ground segment 2, these data are sent to ground segment 1 through dedicated cables.

The vertical architecture is illustrated in Figure 82. This architecture is represented in 3 layers including application layer, data link layer, and physical layer. As the satellite 2 does not need to understand the satellite 1, the application layer only consists data interfaces between the satellite 1 and its ground segment, which are ADCS TLM interface, other TLM interface, and mission data interface. The data link layer represents CCSDS packets between two satellites, and two ground segments. These two CCSDS transport services embedded in onboard computers of two satellites, and in mission control centers of two ground segments must be compatible. Finally, physical layer represents RF links between two satellites, and two ground segments.

Verification of DHS of the data relay satellite system is done with testing compatibility of CCSDS services embedded in two satellites, and testing RF link. Besides, application layer is tested in the same way with MicroDragon satellite.

Verification of compatibility of CCSDS services embedded in two satellites is done by connecting 3 personal computers which are embedded two CCSDS transport services as in Figure 83. This testing configuration is built because normally GEO satellite (the satellite 2) had been launched before starting development of the satellite 1, thus, it is impossible to test with onboard computer of the satellite 2. TLM data are generated from computer 1, and then packed by CCSDS transport service 1 before being sent to computer 2 and computer 3. In computer 3, data packets are de-packed, and displayed. If displayed data in computer 3 are similar to generated data in computer 1, the compatibility of two CCSDS transport services is confirmed.

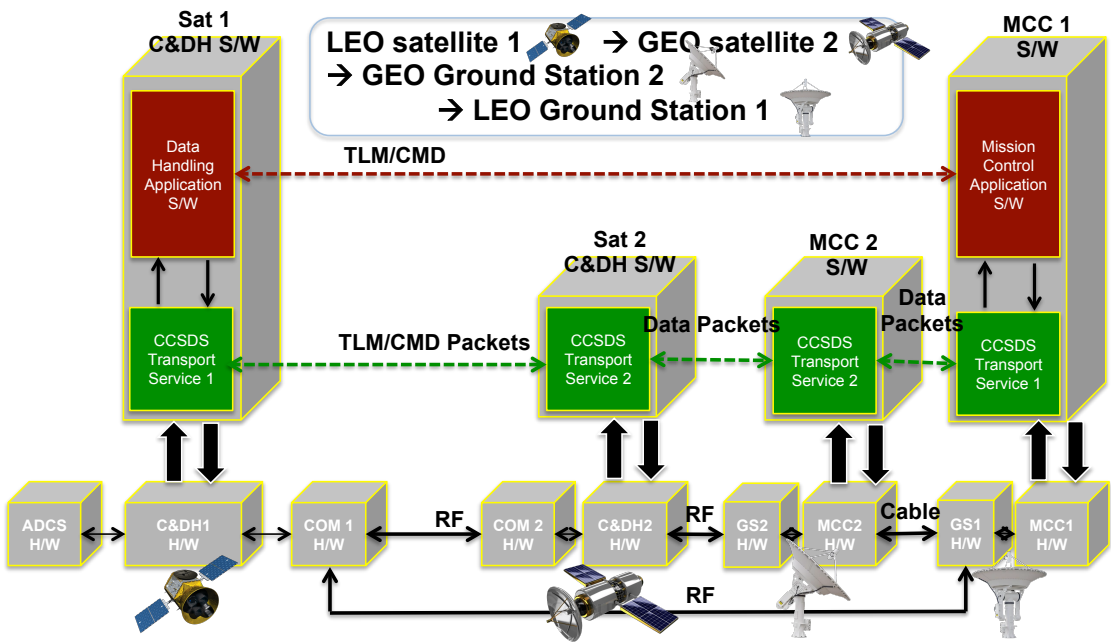


Figure 82. Architecture of DHS of the data relay satellite system

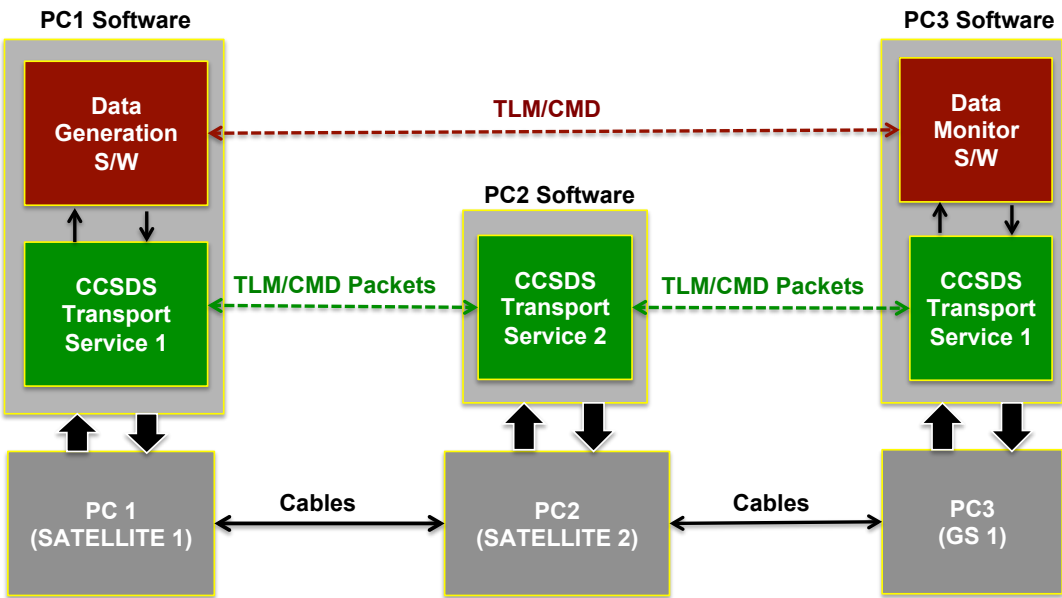


Figure 83. Verification of data link layer of DHS of the data relay satellite system

Verification of RF layer includes testing RF link between 2 satellites, and testing RF link between the satellite 1, and its ground segment.

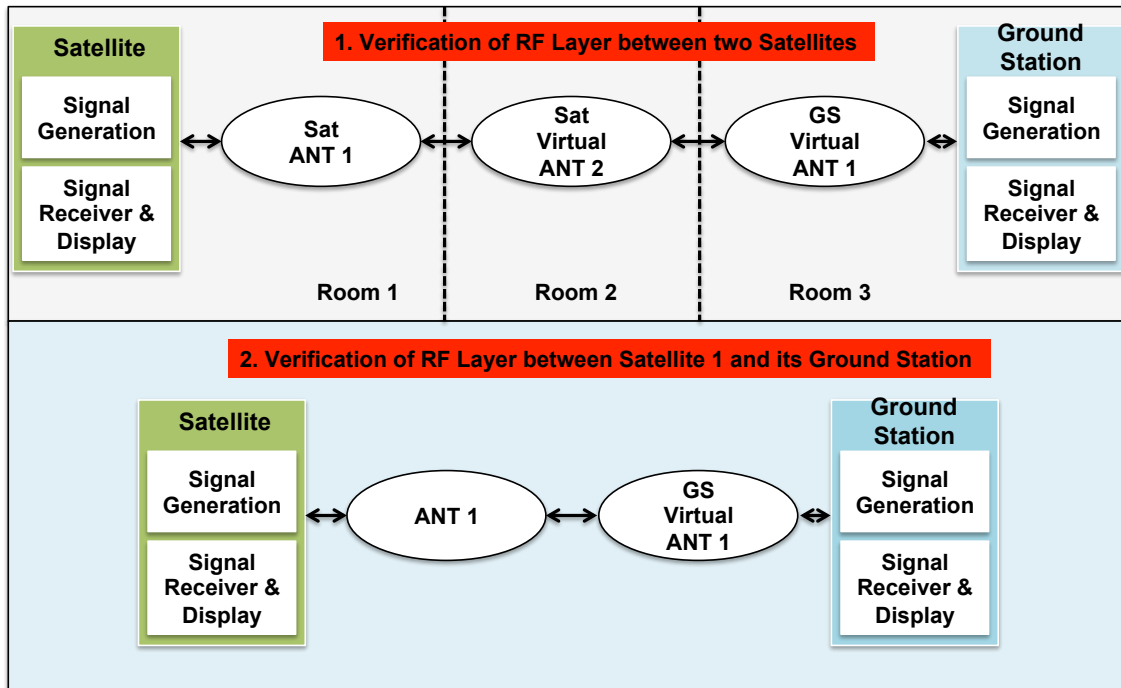


Figure 84. Verification of RF links of DHS of the data relay satellite system

Verification of RF link between two satellites: as GEO satellite is already launched so that testing configuration is built with virtual antenna of the satellite 2. The configuration is arranged as in Figure 84. To verify RF layers of two satellites, antenna of the satellite 1, virtual antenna of satellite 2 and virtual antenna of ground station 1, are put in 3 separate rooms. Between room 1 and room 2, there is a small hole 1, and between room 2 and room 3, there is a small hole 2. The hole 1 and 2 are made in such a way that there is no RF signal transmitted between room 1 and room 3. Input signal is generated from room 1 or room 2, and output signal is displayed in one device in the other of two rooms. If displayed signal is the same with generated signal, RF link between two satellites is confirmed.

Verification of RF link between the satellite 1 and its ground station can be done in the same way with end-to-end test for RF layer in Figure 42. Besides, the verification can also done as in Figure 84. Antenna of the satellite 1 is tested with virtual antenna of ground station 1. If the displayed signal in one end is the same with generated signal in the other end, the RF link between the satellite 1 and its ground station is confirmed.

#### 4.4. Interview with satellite specialists

The interviews are done with five satellite specialists in Tokyo University, and Keio University. Eighteen questions are given to make surveys on understandability, usability and effectiveness of the proposed approach, as described in Table 7.

The following is the results obtained in the interviews including comments and answers of the interviewees.

##### 4.4.1. Interviewee No. 1

The answers of the interviewee No.1 are shown in Table 11. He thought that the idea of the approach was very interesting and impressive. Based on his experience of development of many satellite in Tokyo University, many projects faced problems of increasing development time and cost because iteration designing and testing processes consume a lot of time. He thought that the approach could help to shorten the iteration process but the verification method of the approach should be described in more detail. He suggested that the verification of upper-layer interfaces usually suffered from constraints from the lower-layers, such as data throughputs constraint, time constraint, to name a few, so that to make the verification more reliable, these constraints should be realized.

**Table 11. Interview Results with the interviewee No. 1**

|          |                 |  |
|----------|-----------------|--|
| <b>1</b> | <b>Question</b> | How well do you understand about the satellite data handling system explained and mentioned in the thesis?   |
|          | <b>Answer</b>   | <b>A. I totally understand (90% - 100%)</b><br>B. I almost understand (70% - 90%)<br>C. I fairly understand (40% - 70%)<br>D. I do not understand (<40%)<br>E. Other answers<br><br><b>A</b><br><b>I have been developing many micro/nano/pico-satellites through which I know the system very well.</b> |
| <b>2</b> | <b>Question</b> | How well do you understand about the overall approach to satellite data handling system design for architectural-layer-driven  |

|          |                 |   |
|----------|-----------------|---|
|          |                 | verification?   |
|          | <b>Answer</b>   | <b>A. I totally understand (90% - 100%)</b><br>B. I almost understand (70% - 90%)<br>C. I fairly understand (40% - 70%)<br>D. I do not understand (<40%)<br>E. Other answers<br><br><b>A</b>  |
| <b>3</b> | <b>Question</b> | How well do you understand about horizontal and vertical architecture mentioned in this approach?   |
|          | <b>Answer</b>   | <b>A. I totally understand (90% - 100%)</b><br>B. I almost understand (70% - 90%)<br>C. I fairly understand (40% - 70%)<br>D. I do not understand (<40%)<br>E. Other answers<br><br><b>A</b><br><br><b>I understand the importance of this architecture as well.</b>  |
| <b>4</b> | <b>Question</b> | How well do you understand about verification methods of data handling system mentioned in this approach?   |
|          | <b>Answer</b>   | A. I totally understand (90% - 100%)<br><b>B. I almost understand (70% - 90%)</b><br>C. I fairly understand (40% - 70%)<br>D. I do not understand (<40%)<br>E. Other answers<br><br><b>B</b><br><br><b>I understand the concept and objectives of the proposed system, but how the verification can be done in upper level reliably (such as by simulation? by conceptual analysis? and how confidently the verification results can imply something?) should better be described in more detail.</b> |

|   |                 |   |
|---|-----------------|---|
| 5 | <b>Question</b> | How well do you know about MicroDragon satellite, such as the size, weight, mission and developers of MicroDragon satellite?  |
|   | <b>Answer</b>   | <p><b>A. I totally understand (90% - 100%)</b></p> <p>B. I almost understand (70% - 90%)</p> <p>C. I fairly understand (40% - 70%)</p> <p>D. I do not understand (&lt;40%)</p> <p>E. Other answers</p> <p><b>A</b></p> <p><b>Because I am participating in MDG project.</b></p>                               |
| 6 | <b>Question</b> | How well do you understand about TableSat activities mentioned in this research?  |
|   | <b>Answer</b>   | <p><b>A. I totally understand (90% - 100%)</b></p> <p>B. I almost understand (70% - 90%)</p> <p>C. I fairly understand (40% - 70%)</p> <p>D. I do not understand (&lt;40%)</p> <p>E. Other answers</p> <p><b>A</b></p> <p><b>Because I am preparing table sat in University of Tokyo for MDG project.</b></p> |
| 7 | <b>Question</b> | How easy do you think about applying this approach to design horizontal architecture of data handling system?   |
|   | <b>Answer</b>   | <p>A. It is very easy</p> <p><b>B. It is easy</b></p> <p>C. It is a little bit difficult</p> <p>D. It is very difficult</p> <p>E. Other answers</p> <p><b>B</b></p> <p><b>Conceptually it would be easy to apply, but in order to get good</b></p>  |

|    |                 |  |
|----|-----------------|--|
|    |                 | <b>results by application, the description of the applying process in more detail using “algorithm” type representation.</b>   |
| 8  | <b>Question</b> | How easy do you think about applying this approach to design vertical architecture of data handling system?  |
|    | <b>Answer</b>   | <p>A. It is very easy</p> <p><b>B. It is easy</b></p> <p>C. It is a little bit difficult</p> <p>D. It is very difficult</p> <p>E. Other answers</p> <p><b>B</b></p> <p><b>The same comment as No.7 question. Especially how you can verify the system in upper level lack detailed explanations.</b></p> |
| 9  | <b>Question</b> | How easy do you think about applying this approach to verify data handling system?   |
|    | <b>Answer</b>   | <p>A. It is very easy</p> <p><b>B. It is easy</b></p> <p>C. It is a little bit difficult</p> <p>D. It is very difficult</p> <p>E. Other answers</p> <p><b>B</b></p> <p><b>The same comment as No.7 and No.8.</b></p>   |
| 10 | <b>Question</b> | Do you think that the approach is a good way of design of satellite data handling system?  |
|    | <b>Answer</b>   | <p><b>A. Yes, I think it is a very good way</b></p> <p>B. Yes, I think it is a good way</p> <p>C. No, I do not think it is not a good way</p> <p>D. No, I think it is not good way at all</p> <p>E. Other answers</p> <p><b>A</b></p>  |

|    |                 |  |
|----|-----------------|--|
| 11 | <b>Question</b> | Do you think that horizontal architecture design is useful for verification of data handling system?   |
|    | <b>Answer</b>   | <p><b>A. Yes, I think it is very useful</b></p> <p>B. Yes, I think it is useful</p> <p>C. No, I do not think it is useful</p> <p>D. No, I think it is not useful at all</p> <p>E. Other answers</p> <p><b>A</b></p>  |
| 12 | <b>Question</b> | Do you think that vertical architecture design is useful for verification of data handling system?   |
|    | <b>Answer</b>   | <p><b>A. Yes, I think it is very useful</b></p> <p>B. Yes, I think it is useful</p> <p>C. No, I do not think it is useful</p> <p>D. No, I think it is not useful at all</p> <p>E. Other answers</p> <p><b>A</b></p> <p><b>This is true in the situation that the verification in each layer is really reliable. The problem is how we can make reliable verification in each layer, as the conceptual verification without hardware implementation would be usually very difficult or superficial.</b></p> |
| 13 | <b>Question</b> | Do you think that the approach is a useful way of verification of data handling system?  |
|    | <b>Answer</b>   | <p><b>A. Yes, I think it is a very useful way</b></p> <p>B. Yes, I think it is an useful way</p> <p>C. No, I do not think it is an useful way</p> <p>D. No, I think it is not an useful way</p> <p>E. Other answers</p> <p><b>A</b></p>  |

|    |                 |  |
|----|-----------------|--|
|    |                 | <b>This is also true in condition that the verification is reliable.</b>   |
| 14 | <b>Question</b> | Do you think that horizontal architecture design of data handling system of MicroDragon satellite is good?   |
|    | <b>Answer</b>   | <b>A. Yes, I think it is very good</b><br>B. Yes, I think it is good<br>C. No, I do not think it is good<br>D. No, I think it is not good at all<br>E. Other answers<br><br><b>A</b><br><b>I encourage you to apply this method to actual MDG project.</b> |
| 15 | <b>Question</b> | Do you think that vertical architecture design of data handling system of MicroDragon satellite is good?   |
|    | <b>Answer</b>   | <b>A. Yes, I think it is very good</b><br>B. Yes, I think it is good<br>C. No, I do not think it is good<br>D. No, I think it is not good at all<br>E. Other answers<br><br><b>A</b><br><b>I encourage you to apply this method to actual MDG project</b>  |
| 16 | <b>Question</b> | Do you think that applying this approach to design and verification of data handling system of MicroDragon satellite can reduce development risks, especially for development of a new system ?  |
|    | <b>Answer</b>   | <b>A. Yes, I think so</b><br>B. No, I do not think so<br>C. Other answers<br><br><b>A</b><br><b>This is true in condition that the verification in each layer is reliable.</b>   |
| 17 | <b>Question</b> | Do you think that the approach is a beneficial way of design and   |

|           |                 |  |
|-----------|-----------------|--|
|           |                 | verification of data handling system of a complex satellite system, such as a data relay satellite system, in terms of reduction of development risks?           |
|           | <b>Answer</b>   | <b>A. Yes, I think so</b><br>B. No, I do not think so<br><br><b>A</b><br><b>We need a general methodology to carry out reliable verifications in each layer.</b> |
| <b>18</b> | <b>Question</b> | Do you think that the approach has some advantages over some other approaches mentioned in this thesis, in terms of reduction of development risks?              |
|           | <b>Answer</b>   | <b>A. Yes, I think so</b><br>B. No, I do not think so<br><br><b>A</b>  |

#### 4.4.2. Interviewee No.2

The answers of the interviewee No.2 is shown in Table 12. He is very interested in the approach. He thought it was difficult to share design concept among team members, so that architectural-layers visualization in this approach would be very helpful for improving mutual understanding. He strongly recommended that the approach should be applied to an actual satellite development project.

Table 12. Interview Results with the interviewee No. 2

|          |                 |   |
|----------|-----------------|---|
| <b>1</b> | <b>Question</b> | How well do you understand about the satellite data handling system explained and mentioned in the thesis?  |
|          | <b>Answer</b>   | <b>A. I totally understand (90% - 100%)</b><br>I think your explanation of “CDH subsystem” in your thesis is well-summarized and good quality and quantity. |
| <b>2</b> | <b>Question</b> | How well do you understand about the overall approach to satellite data handling system design for architectural-layer-driven verification?                 |
|          | <b>Answer</b>   | <b>A. I totally understand (90% - 100%)</b>   |

|   |                 |  |
|---|-----------------|--|
|   |                 | <p>I think your explanation of “architectural-layer-driven verification” in your thesis is good quality and quantity. By reading your thesis, I became interested in this approach. This approach is very useful for sharing the designers’ vision with the members of a project team appropriately.</p> <p>As you may know, it’s very difficult to share the design concept among a project team, and “misunderstanding of the design” sometimes causes a critical trouble in the project. The visualization of “architectural-layer” is very helpful for better understanding among a team.</p> <p>I strongly think that I would like to introduce and adopt this approach my actual project work for effective development.</p> |
| 3 | <b>Question</b> | How well do you understand about horizontal and vertical architecture mentioned in this approach?  |
|   | <b>Answer</b>   | <p><b>B. I almost understand (70% - 90%)</b></p> <p>I think your explanation of “horizontal and vertical architecture” in your thesis is well-summarized and good quality. For better understanding, I would like to consider some examples based on my project experience by myself.</p>  |
| 4 | <b>Question</b> | How well do you understand about verification methods of data handling system mentioned in this approach?  |
|   | <b>Answer</b>   | <p><b>B. I almost understand (70% - 90%)</b></p> <p>I think your explanation of “verification methods of CDH” in your thesis is well-summarized and good quality.</p>  |
| 5 | <b>Question</b> | How well do you know about MicroDragon satellite, such as the size, weight, mission and developers of MicroDragon satellite?   |
|   | <b>Answer</b>   | <p><b>E. Other answers</b></p> <p>I’m also a faculty member of MicroDragon Project as a mentor of EPS.</p> <p>Your explanations and topics of MicroDragon Project are good quality.</p>  |
| 6 | <b>Question</b> | How well do you understand about TableSat activities mentioned in  |

|    |                 |   |
|----|-----------------|---|
|    |                 | this research?  |
|    | <b>Answer</b>   | <b>E. Other answers</b><br>I'm also a faculty member of MicroDragon Project as a mentor of EPS.<br>Your explanations and topics of TableSat activities of MicroDragon Project are good quality. |
| 7  | <b>Question</b> | How easy do you think about applying this approach to design horizontal architecture of data handling system?   |
|    | <b>Answer</b>   | <b>B. It is easy</b>  |
| 8  | <b>Question</b> | How easy do you think about applying this approach to design vertical architecture of data handling system?   |
|    | <b>Answer</b>   | <b>B. It is easy</b><br>I think "horizontal and vertical architecture" is suitable for CDH design.  |
| 9  | <b>Question</b> | How easy do you think about applying this approach to verify data handling system?  |
|    | <b>Answer</b>   | <b>B. It is easy</b>  |
| 10 | <b>Question</b> | Do you think that the approach is a good way of design of satellite data handling system?   |
|    | <b>Answer</b>   | <b>A. Yes, I think it is a very good way</b>  |
| 11 | <b>Question</b> | Do you think that horizontal architecture design is useful for verification of data handling system?  |
|    | <b>Answer</b>   | <b>A. Yes, I think it is very useful</b>  |
| 12 | <b>Question</b> | Do you think that vertical architecture design is useful for verification of data handling system?  |
|    | <b>Answer</b>   | <b>A. Yes, I think it is very useful</b>  |
| 13 | <b>Question</b> | Do you think that the approach is a useful way of verification of data handling system?   |
|    | <b>Answer</b>   | <b>A. Yes, I think it is a very useful way</b>  |
| 14 | <b>Question</b> | Do you think that horizontal architecture design of data handling system of MicroDragon satellite is good?  |
|    | <b>Answer</b>   | <b>A. Yes, I think it is very good</b>  |

|    |                 |   |
|----|-----------------|---|
| 15 | <b>Question</b> | Do you think that vertical architecture design of data handling system of MicroDragon satellite is good?  |
|    | <b>Answer</b>   | <b>A. Yes, I think it is very good</b>  |
| 16 | <b>Question</b> | Do you think that applying this approach to design and verification of data handling system of MicroDragon satellite can reduce development risks, especially for development of a new system?                          |
|    | <b>Answer</b>   | <b>A. Yes, I think so</b>   |
| 17 | <b>Question</b> | Do you think that the approach is a beneficial way of design and verification of data handling system of a complex satellite system, such as a data relay satellite system, in terms of reduction of development risks? |
|    | <b>Answer</b>   | <b>A. Yes, I think so</b>   |
| 18 | <b>Question</b> | Do you think that the approach has some advantages over some other approaches mentioned in this thesis, in terms of reduction of development risks?   |
|    | <b>Answer</b>   | <b>A. Yes, I think so</b>   |

#### 4.4.3. Interviewee No.3

The answers of the interviewee No. 3 are described in Table 13. He suggested that the approach should be explained systematically in more details. In his opinions, the objective of the research was good, and the approach could support design and verification of satellite data handling system. However, description of the approach should be improved by giving some actual examples in actual satellite development projects, which showed how the development risks were reduced.

**Table 13. Interview Results with the interviewee No. 3**

|   |                 |  |
|---|-----------------|--|
| 1 | <b>Question</b> | How well do you understand about the satellite data handling system explained and mentioned in the thesis?   |
|   | <b>Answer</b>   | <b>A. I totally understand (90% - 100%)</b><br>B. I almost understand (70% - 90%)<br>C. I fairly understand (40% - 70%)<br>D. I do not understand (<40%)<br>E. Other answers |

|   |                 |  |
|---|-----------------|--|
| 2 | <b>Question</b> | How well do you understand about the overall approach to satellite data handling system design for architectural-layer-driven verification?                                  |
|   | <b>Answer</b>   | <b>A. I totally understand (90% - 100%)</b><br>B. I almost understand (70% - 90%)<br>C. I fairly understand (40% - 70%)<br>D. I do not understand (<40%)<br>E. Other answers |
| 3 | <b>Question</b> | How well do you understand about horizontal and vertical architecture mentioned in this approach?  |
|   | <b>Answer</b>   | <b>A. I totally understand (90% - 100%)</b><br>B. I almost understand (70% - 90%)<br>C. I fairly understand (40% - 70%)<br>D. I do not understand (<40%)<br>E. Other answers |
| 4 | <b>Question</b> | How well do you understand about verification methods of data handling system mentioned in this approach?  |
|   | <b>Answer</b>   | <b>A. I totally understand (90% - 100%)</b><br>B. I almost understand (70% - 90%)<br>C. I fairly understand (40% - 70%)<br>D. I do not understand (<40%)<br>E. Other answers |
| 5 | <b>Question</b> | How well do you know about MicroDragon satellite, such as the size, weight, mission and developers of MicroDragon satellite?   |
|   | <b>Answer</b>   | <b>A. I totally understand (90% - 100%)</b><br>B. I almost understand (70% - 90%)<br>C. I fairly understand (40% - 70%)<br>D. I do not understand (<40%)<br>E. Other answers |
| 6 | <b>Question</b> | How well do you understand about TableSat activities mentioned in this research?   |
|   | <b>Answer</b>   | <b>A. I totally understand (90% - 100%)</b>  |

|   |                 |  |
|---|-----------------|--|
|   |                 | B. I almost understand (70% - 90%)<br>C. I fairly understand (40% - 70%)<br>D. I do not understand (<40%)<br>E. Other answers  |
| 7 | <b>Question</b> | How easy do you think about applying this approach to design horizontal architecture of data handling system?  |
|   | <b>Answer</b>   | A. It is very easy<br><b>B. It is easy</b><br>C. It is a little bit difficult<br>D. It is very difficult<br>E. Other answers<br><br><b>In order to make easier application of your method, you should break down your method into small steps.</b> |
| 8 | <b>Question</b> | How easy do you think about applying this approach to design vertical architecture of data handling system?  |
|   | <b>Answer</b>   | A. It is very easy<br><b>B. It is easy</b><br>C. It is a little bit difficult<br>D. It is very difficult<br>E. Other answers<br><br><b>Same of No.7</b>  |
| 9 | <b>Question</b> | How easy do you think about applying this approach to verify data handling system?   |
|   | <b>Answer</b>   | A. It is very easy<br><b>B. It is easy</b><br>C. It is a little bit difficult<br>D. It is very difficult<br>E. Other answers<br><br><b>Same of No.7</b>  |

|    |                 |   |
|----|-----------------|---|
| 10 | <b>Question</b> | Do you think that the approach is a good way of design of satellite data handling system?   |
|    | <b>Answer</b>   | <p>A. Yes, I think it is a very good way</p> <p><b>B. Yes, I think it is a good way</b></p> <p>C. No, I do not think it is not a good way</p> <p>D. No, I think it is not good way at all</p> <p>E. Other answers</p> <p><b>Your objectives and approach seem to be very good. Please continue to improve your method through applying to actual satellite project.</b></p> |
| 11 | <b>Question</b> | Do you think that horizontal architecture design is useful for verification of data handling system?  |
|    | <b>Answer</b>   | <p>A. Yes, I think it is very useful</p> <p><b>B. Yes, I think it is useful</b></p> <p>C. No, I do not think it is useful</p> <p>D. No, I think it is not useful at all</p> <p>E. Other answers</p>   |
| 12 | <b>Question</b> | Do you think that vertical architecture design is useful for verification of data handling system?  |
|    | <b>Answer</b>   | <p>A. Yes, I think it is very useful</p> <p><b>B. Yes, I think it is useful</b></p> <p>C. No, I do not think it is useful</p> <p>D. No, I think it is not useful at all</p> <p>E. Other answers</p> <p><b>Since you have divided architecture into three layers, the prospect of the verification seems to be enhanced.</b></p>   |
| 13 | <b>Question</b> | Do you think that the approach is a useful way of verification of data handling system?   |
|    | <b>Answer</b>   | <p>A. Yes, I think it is a very useful way</p> <p><b>B. Yes, I think it is a useful way</b></p>   |

|    |                 |   |
|----|-----------------|---|
|    |                 | <p>C. No, I do not think it is an useful way</p> <p>D. No, I think it is not an useful way</p> <p>E. Other answers</p>  |
| 14 | <b>Question</b> | Do you think that horizontal architecture design of data handling system of MicroDragon satellite is good?  |
|    | <b>Answer</b>   | <p>A. Yes, I think it is very good</p> <p><b>B. Yes, I think it is good</b></p> <p>C. No, I do not think it is good</p> <p>D. No, I think it is not good at all</p> <p>E. Other answers</p>   |
| 15 | <b>Question</b> | Do you think that vertical architecture design of data handling system of MicroDragon satellite is good?  |
|    | <b>Answer</b>   | <p><b>A. Yes, I think it is very good</b></p> <p>B. Yes, I think it is good</p> <p>C. No, I do not think it is good</p> <p>D. No, I think it is not good at all</p> <p>E. Other answers</p> <p><b>Generally speaking, your vertical architecture design seems to be very good. However, more detail process and a few more examples must be shown so that everybody can easily understand the method.</b></p> |
| 16 | <b>Question</b> | Do you think that applying this approach to design and verification of data handling system of MicroDragon satellite can reduce development risks, especially for development of a new system?  |
|    | <b>Answer</b>   | <p><b>A. Yes, I think so</b></p> <p>B. No, I do not think so</p> <p>C. Other answers</p> <p><b>If possible, please show some examples of the result that shows the reduction of development risks.</b></p>  |
| 17 | <b>Question</b> | Do you think that the approach is a beneficial way of design and  |

|           |                 |  |
|-----------|-----------------|--|
|           |                 | verification of data handling system of a complex satellite system, such as a data relay satellite system, in terms of reduction of development risks?   |
|           | <b>Answer</b>   | <b>A. Yes, I think so</b><br>B. No, I do not think so<br><br><b>If possible, please show some examples of the result that shows the reduction or development risks.</b>  |
| <b>18</b> | <b>Question</b> | Do you think that the approach has some advantages over some other approaches mentioned in this thesis, in terms of reduction of development risks?  |
|           | <b>Answer</b>   | <b>A. Yes, I think so</b><br>B. No, I do not think so<br><br><b>I believe that this approach has advantages. If possible, please compare to some existing approaches and show the result. This must be very helpful for those who want to use this approach.</b> |

#### 4.4.4. Interviewee No.4

The answers of the interviewee No. 4 are shown in Table 14. In his opinions, operational design should be vertical architecture, instead of horizontal architecture. So that he thought that description of the approach was not clear enough to understand how to design and verify satellite data handling system. However, in his point of view, the concept of testing upper-layers interfaces at low-level testing was a great idea to reduce development risks by shortening iteration development processes. He suggested that the approach should be explained more systematically so that it would be easier for satellite developers to design and test data handling system. Besides, using colors to represent interface layers should be identical from figures to figures to help others have better understanding.

**Table 14. Interview results with the interviewee No. 4**

|          |                 |  |
|----------|-----------------|--|
| <b>1</b> | <b>Question</b> | How well do you understand about the satellite data handling system explained and mentioned in the thesis? |
|----------|-----------------|--|

|   |                 |  |
|---|-----------------|--|
|   | <b>Answer</b>   | <b>B. I almost understand (70% - 90%)</b>  |
| 2 | <b>Question</b> | How well do you understand about the overall approach to satellite data handling system design for architectural-layer-driven verification?  |
|   | <b>Answer</b>   | <b>B. I almost understand (70% - 90%)</b>  |
| 3 | <b>Question</b> | How well do you understand about horizontal and vertical architecture mentioned in this approach?  |
|   | <b>Answer</b>   | <b>C. I fairly understand (40% - 70%)</b>  |
| 4 | <b>Question</b> | How well do you understand about verification methods of data handling system mentioned in this approach?  |
|   | <b>Answer</b>   | <b>B. I almost understand (70% - 90%)</b>  |
| 5 | <b>Question</b> | How well do you know about MicroDragon satellite, such as the size, weight, mission and developers of MicroDragon satellite?   |
|   | <b>Answer</b>   | <b>A. I totally understand (90% - 100%)</b>  |
| 6 | <b>Question</b> | How well do you understand about TableSat activities mentioned in this research?   |
|   | <b>Answer</b>   | <b>A. I totally understand (90% - 100%)</b>  |
| 7 | <b>Question</b> | How easy do you think about applying this approach to design horizontal architecture of data handling system?  |
|   | <b>Answer</b>   | <b>C. It is a little bit difficult</b><br>It may be not so clear for a satellite developing team to exactly identify “horizontal architecture” of their design.                        |
| 8 | <b>Question</b> | How easy do you think about applying this approach to design vertical architecture of data handling system?  |
|   | <b>Answer</b>   | <b>C. It is a little bit difficult</b><br>It may be not so clear for a satellite developing team to exactly identify “vertical architecture” of their design.                          |
| 9 | <b>Question</b> | How easy do you think about applying this approach to verify data handling system?   |
|   | <b>Answer</b>   | <b>C. It is a little bit difficult</b><br>It may be easy if the satellite under consideration is similar to the one described in this article but if not it requires much of tailoring |

|    |                 |   |
|----|-----------------|---|
|    |                 | and it can be quite difficult.  |
| 10 | <b>Question</b> | Do you think that the approach is a good way of design of satellite data handling system?   |
|    | <b>Answer</b>   | <b>A. Yes, I think it is a very good way</b>  |
| 11 | <b>Question</b> | Do you think that horizontal architecture design is useful for verification of data handling system?  |
|    | <b>Answer</b>   | <b>E. Other answers</b><br>Yes, if and only if one who is planning verification fully understand the horizontal architecture.   |
| 12 | <b>Question</b> | Do you think that vertical architecture design is useful for verification of data handling system?  |
|    | <b>Answer</b>   | <b>E. Other answers</b><br>Yes, if and only if one who is planning verification fully understand the vertical architecture.   |
| 13 | <b>Question</b> | Do you think that the approach is a useful way of verification of data handling system?   |
|    | <b>Answer</b>   | <b>E. Other answers</b><br>Yes, if and only if one who is planning verification fully understand the vertical and horizontal architecture.                                |
| 14 | <b>Question</b> | Do you think that horizontal architecture design of data handling system of MicroDragon satellite is good?  |
|    | <b>Answer</b>   | <b>E. Other answers</b><br>In terms of data transmission and receiving it is Yes.<br>But the answer may be different if data usage use cases are provided in more detail. |
| 15 | <b>Question</b> | Do you think that vertical architecture design of data handling system of MicroDragon satellite is good?  |
|    | <b>Answer</b>   | <b>E. Other answers</b><br>In terms of data transmission and receiving it is Yes.<br>But the answer may be different if data usage use cases are provided in more detail. |
| 16 | <b>Question</b> | Do you think that applying this approach to design and verification   |

|           |                 |   |
|-----------|-----------------|---|
|           |                 | of data handling system of MicroDragon satellite can reduce development risks, especially for development of a new system?  |
|           | <b>Answer</b>   | <b>C. Other answers</b><br>Yes, if and only if development team can derive the action plans according to the design and verification plan.  |
| <b>17</b> | <b>Question</b> | Do you think that the approach is a beneficial way of design and verification of data handling system of a complex satellite system, such as a data relay satellite system, in terms of reduction of development risks? |
|           | <b>Answer</b>   | <b>A. Yes, I think so</b><br>If and only if development team fully understands both horizontal and vertical architecture and also understands fundamental concept of verification and validation.                       |
| <b>18</b> | <b>Question</b> | Do you think that the approach has some advantages over some other approaches mentioned in this thesis, in terms of reduction of development risks?   |
|           | <b>Answer</b>   | <b>A. Yes, I think so</b><br>If and only if development team fully understands both horizontal and vertical architecture and also understands fundamental concept of verification and validation.                       |

#### 4.4.5. Interviewee No.5

The answers of the interviewee No. 5 are described in Table 15. He had a lot of experience in development of satellite. He said that the philosophy of the approach has been applied to some satellites. However, the systematical description of development of satellite data handling system in this approach creates its originality. He stated that the approach could help to reduce risks of increasing development time and cost, and this fact has been proved through some actual satellite development projects. He also recommended adding end-to-end test for verification of RF link, which was often used in space industry.

**Table 15. Interview results with the interviewee No. 5**

|          |                 |  |
|----------|-----------------|--|
| <b>1</b> | <b>Question</b> | How well do you understand about the satellite data handling system explained and mentioned in the thesis? |
|----------|-----------------|--|

|   |                 |   |
|---|-----------------|---|
|   | <b>Answer</b>   | <b>A. I totally understand (90% - 100%)</b><br>I totally understand because I have been working in space industry for more than 24 years.   |
| 2 | <b>Question</b> | How well do you understand about the overall approach to satellite data handling system design for architectural-layer-driven verification?   |
|   | <b>Answer</b>   | <b>A. I totally understand (90% - 100%)</b><br>Same as above.   |
| 3 | <b>Question</b> | How well do you understand about horizontal and vertical architecture mentioned in this approach?   |
|   | <b>Answer</b>   | <b>A. I totally understand (90% - 100%)</b><br>Same as above  |
| 4 | <b>Question</b> | How well do you understand about verification methods of data handling system mentioned in this approach?   |
|   | <b>Answer</b>   | <b>A. I totally understand (90% - 100%)</b><br>Same as above.   |
| 5 | <b>Question</b> | How well do you know about MicroDragon satellite, such as the size, weight, mission and developers of MicroDragon satellite?  |
|   | <b>Answer</b>   | <b>A. I totally understand (90% - 100%)</b><br>Same as above.   |
| 6 | <b>Question</b> | How well do you understand about TableSat activities mentioned in this research?  |
|   | <b>Answer</b>   | <b>A. I totally understand (90% - 100%)</b><br>Same as above.   |
| 7 | <b>Question</b> | How easy do you think about applying this approach to design horizontal architecture of data handling system?   |
|   | <b>Answer</b>   | <b>E. Other answers</b><br>The approach is generally applied to satellite development in space companies. However, I think the originality of this approach lies at the fact that the approach clarifies a systematical process with vertical and horizontal architecture which has yet to be introduced in any researches. |
| 8 | <b>Question</b> | How easy do you think about applying this approach to design vertical architecture of data handling system?   |
|   | <b>Answer</b>   | <b>E. Other answers</b><br>Same as No. 7  |
| 9 | <b>Question</b> | How easy do you think about applying this approach to verify data   |

|           |                 |   |
|-----------|-----------------|---|
|           |                 | handling system?  |
|           | <b>Answer</b>   | <b>E. Other answers</b><br>Same as No. 7  |
| <b>10</b> | <b>Question</b> | Do you think that the approach is a good way of design of satellite data handling system?   |
|           | <b>Answer</b>   | <b>B. Yes, I think it is a good way</b>   |
| <b>11</b> | <b>Question</b> | Do you think that horizontal architecture design is useful for verification of data handling system?  |
|           | <b>Answer</b>   | <b>A. Yes, I think it is very useful</b><br>Satellite developers have to do that.   |
| <b>12</b> | <b>Question</b> | Do you think that vertical architecture design is useful for verification of data handling system?  |
|           | <b>Answer</b>   | <b>A. Yes, I think it is very useful</b><br>Satellite developers have to do that.   |
| <b>13</b> | <b>Question</b> | Do you think that the approach is a useful way of verification of data handling system?   |
|           | <b>Answer</b>   | <b>A. Yes, I think it is a very useful way</b><br>Same as No. 12  |
| <b>14</b> | <b>Question</b> | Do you think that horizontal architecture design of data handling system of MicroDragon satellite is good?  |
|           | <b>Answer</b>   | <b>B. Yes, I think it is good</b>   |
| <b>15</b> | <b>Question</b> | Do you think that vertical architecture design of data handling system of MicroDragon satellite is good?  |
|           | <b>Answer</b>   | <b>B. Yes, I think it is good</b>   |
| <b>16</b> | <b>Question</b> | Do you think that applying this approach to design and verification of data handling system of MicroDragon satellite can reduce development risks, especially for development of a new system?                          |
|           | <b>Answer</b>   | <b>A. Yes, I think so</b><br>Philosophy of the approach has been applied to few satellites. Reduction of development risks was proved in actual satellite development projects.   |
| <b>17</b> | <b>Question</b> | Do you think that the approach is a beneficial way of design and verification of data handling system of a complex satellite system, such as a data relay satellite system, in terms of reduction of development risks? |
|           | <b>Answer</b>   | <b>A. Yes, I think so</b><br>Same as No. 16   |

---



---

|    |                 |  |
|----|-----------------|--|
| 18 | <b>Question</b> | Do you think that the approach has some advantages over some other approaches mentioned in this thesis, in terms of reduction of development risks?  |
|    | <b>Answer</b>   | <b>A. Yes, I think so</b><br>It is a good approach because the philosophy of the approach has been applied to satellite development in general. And I think the originality of this approach is the fact that the approach clarifies a systematical process with vertical and horizontal architecture, which has yet to be introduced in any researches. |

#### 4.5. Discussion and conclusion on results of evaluation of the approach

##### 4.5.1. Results of applying the approach to develop DHS of MicroDragon satellite

Architecture of DHS of MicroDragon satellite is designed based on the approach. In opinions of all the interviewed satellite specialists, the result of architecture design is good.

MicroDragon satellite has not integrated yet. Currently, engineering model of some component is available. Besides, the ground segment has yet to be developed. For those reasons, it seems impossible to test telemetry and command in application layer at the moment. However, by applying the approach, a simple testing configuration for verification of telemetry and command in application layer was built based on onboard computer and onboard software of MicroDragon, and TableSat facilities at Tokyo University. The testing results can confirm data interfaces in application layer, data link layer, and RF layer. The confirmation of data interfaces in this early phase of development of MicroDragon satellite is very useful for saving development time and cost. If the confirmation is postponed until the satellite is integrated, and some unexpected problems are detected during system-level tests, it will consume a lot of time to de-integrate the satellite, identify and fix the problems, and then re-integrate the satellite.

Success of identifying & testing application layer with CMD & TLM of data handling system of MicroDragon satellite based on vertical architecture design following the approach, proves testability of upper-layers before integration with the approach.

#### ***4.5.2. Results of applying the approach to develop DHS of a data relay satellite system***

The data relay satellite system is a complex system of systems. Confirmation of mutual understanding among systems is challenging yet necessary because of difficulties of identifying how data interfaces in each layer can be verified. As this approach supports visualization of architectural layers, it offers easier way to imagine how data handling system can be tested in each layer, as it is seen in **Error! Reference source not found.** With the help of architecture design, it is possible to confirm upper-layers interfaces at low-level testing, even for a complex system like the data relay satellite system.

During interviews with some satellite specialists, they were impressive with design of data handling system of the data relay satellite system, and they thought the approach was a useful way to design and verify architecture of the complex satellite system, as architectural-layer visualization in the approach makes it easier to understand the system.

Representation of upper-layers interfaces in data handling system architecture of complex systems, such as data relay satellite system, shows the importance and efficiency of the approach.

#### ***4.5.3. Results of interviewing satellite specialists***

The interview results are shown in Table 11, Table 12, Table 13, Table 14, and Table 15. The differences of the results show different viewpoints of the interviewees. The viewpoint is decided by personal opinions, background of interviewees, and situations occurred during the interviews. The actual satellite developers tend to focus on applying the approach to actual satellite development projects, and design of data handling system of an actual satellite, and realization of verification configuration with constraints from hardware and software, for example how to realize time response, and distortion of short pulses in hardware components with software simulators. While system engineering people tend to concentrate on applying system engineering to design the approach, and to describe the approach in a systematical way, to make designing and verifying processes to be traceable and identical. However, in general, all the interviewees found that the approach was interesting, applicable, and had the originality and some advantages over previous approaches in terms of reducing development risks.

Besides, they also agreed that the design of data handling system of MicroDragon satellite and the data relay satellite system by applying the approach were good.

Based on interview results, it can be concluded that the approach is highly applicable, effective and objective achievable.

In conclusion, the evaluation results show that the approach works and supports verification of layered interfaces in early phases, which allows to reduce iteration testing processes and development risks.

## **Chapter 5 Conclusion**

### **5.1. Summary**

The research has introduced a new approach to satellite data handling system for architectural-layer-driven verification. The approach consists of two parts including design of architecture of DHS, and verification of DHS. To support verification of upper-layers interfaces at low-level testing, two types of architecture call “horizontal” & “vertical” architecture are designed. The horizontal architecture is an architecture designed in operational domain and systems domain by using operational view and systems view of DoDAF. The vertical architecture is designed with vertical viewpoint to show architectural-layers in the system.

Evaluation of the approach is done by applying the approach to MicroDragon satellite, a data relay satellite system, and interviewing 5 satellite specialists from universities and space industry.

Evaluation results show that the approach works, has originality and high applicability. Besides, it is understandable, usable, effective, and objective-achievable.

The approach supports in confirming understandings among subsystems in a system, or among systems in a system of systems yet it also causes number of tests increase, which results in growth of development cost. Therefore, the approach should only be applied to development of a new data handling system where early confirmation of understandings among subsystems is very important, rather than be applied to production line.

### **5.2. Future work**

As the proposed approach focuses on designing process of data handling system rather than verification configuration which is already stable and saturated in space industry, performance verification of data handling system which is actually out of scope of the current research is yet to be mentioned in the approach. Performance verification can be done in future as extension of the approach.

---

---

## References

- [1] Sandra May (2015) *NASA knows: Rockets and Space Vehicles (Grades 5-8)*, Available at: [www.nasa.gov](http://www.nasa.gov) (Accessed: May 1st, 2016).
- [2] (2015) *A Comprehensive Census of All Spacecraft Ever Launched*, Available at: <http://claudelafleur.qc.ca> (Accessed: May 1st, 2016).
- [3] Gene Crandall, Chris Abbott, Scott Reedy (2007) 'Evolution of a Standardized Repeatable Satellite Test Approach', *IEEE Autotestcon*, (1088-7725), pp. 435-444.
- [4] Jochen Wolle (2013) 'LXI in Satellite System Testing', *IEEE Autotestcon*, (1088-7725), pp. 1-5.
- [5] J. Eickhoff (2012) *Onboard Computers, Onboard Software and Satellite Operations*. [Online]. Available at: <http://www.springer.com/> (Accessed: Mar. 12th, 2015).
- [6] D.I. Dalgleish (1989) *An Introduction to Satellite Communications*, : IET.
- [7] Jen Eickhoff, Joerg Flemming (2002) 'Model-based Development and Verification', *Proceedings of DASIA 2002, 13 - 16 May 2002, Dublin, Ireland*, (92-9092-819-0), pp. 11.1 - 11.14.
- [8] M. Bozzano, R. Cavada, A. Cimatti, J.-P. Katoen, V.Y. Nguyen, T. Noll, X. Olive (2009) *Formal Verification and Validation of AADL Models*, Available at: <http://compass.informatik.rwth-aachen.de> (Accessed: May 1st, 2016).
- [9] Peter H. Feiler, David P. Gluch, John J. Hudak (2006) *The architecture Analysis and Design Language (AADL): An Introduction*, <http://www.sei.cmu.edu>: Carnegie Mellon University.
- [10] Annalisa L. Weigel (2000) *Spacecraft System-level Integration and Test Discrepancies: Characterizing Distributions and Costs*, Massachusetts Institute of Technology: .
- [11] B. Vaartjes, H.J. Hamann, R. Amini (2007) 'Integration and Verification of a Command and Data Handling Subsystem for Nano-Satellite Projects with Critical Time Constraints: Delfi-C3', *The 58th International Astronautical Congress*.
- [12] Andrew Strange (2006) *Communication On Board: A Satellite Data Handling and Ground Communication System*, <http://digitalcommons.usu.edu>: Taylor University.
- [13] Pablito B. Yra, Matteo Genna, Sean McMahon, Kevin Kerns (2010) 'Next-generation Spacecraft Command and Data Handling System based on RAD750 Processor', *The 28th AIAA International Communications Satellite Systems Conference*,

---

(AIAA 2010-8888).

[14] D.V. Ramana, Jolie. R (2012) *Satellite Data Transmitting Systems and In-orbit Performance*. [Online]. Available at: <http://www.highfrequencyelectronics.com> (Accessed: May 1st, 2016).

[15] Marie-Aude Esteve, Joost-Pieter Katoen, Viet Yen Nguyen, Bart Postma, Yuri Yushtein (2012) 'Formal Correctness, Safety, Dependability, and Performance Analysis of a Satellite', *IEEE*, The 34th International Conference on Software Engineering (978-1-4673-1065-9).

[16] K.H. Jeeja, Chayan Dutta, Lalitkrushna Thakar, P.S. Sura, M. Vanitha (2012) 'Test Station for CCSDS based on Data Handling Systems of Mini Satellites', *IEEE*, 2012 IEEE International Conference (978-1-4673-2045-0).

[17] William F. Tosney, Steve Pavlica (2004) 'Satellite Verification Planning: Best Practices and Pitfalls Related to Testing', *Proceedings of the 5th International Symposium on Environmental Testing for Space Programmes*.

[18] Stephen A. Jacklin (2015) *Survey of Verification and Validation Techniques for Small Satellite Software Development*, <http://ntrs.nasa.gov>: 2015 Space Tech Expo Conference.

[19] George Ritter (2002) *Software Development and Test Methodology for a Distributed Ground System*, : Bibliogov.

[20] Anna Maria Ambrosi, Fatima Mattiello-Francisco, Eliane Martins (2008) 'An Independent Software Verification and Validation Process for Space Applications', *SpaceOps 2008 Conference*, (AIAA 2008-3517).

[21] Jeremie Pouly, Sylvain Jouanneau (2012) *Model-based Specification of the Flight Software of an Autonomous Satellite*, Available at: <http://web1.see.asso.fr> (Accessed: May 1st, 2016).

[22] Brett Murphy, Amory Wakefield, Jon Friedman (2008) *Best Practices for Verification, Validation, and Test in Model-based Design*, The Mathworks, Inc.

[23] Sarah E. McNeese, David A. Spencer (2011) *Center for Space Systems Mission Success Assurance: Requirement Verification and Spacecraft Integration and Test Program*, Georgia Institute of Technology.

[24] Department of Defense (2004) *DoD Architecture Framework. Version 1.0* [Online]. Available at: <http://www.opengroup.org> (Accessed: Mar. 26th, 2015).

[25] Julia D. White (2010) *Test Like You Fly: Assessment and Implementation Process*,

---

Space Systems Group.

[26] V.S. Bagad (2009) *Satellite Communications*, : Technical Publications.

[27] James R. Wertz (2011) *Space Mission Engineering: the new SMAD*, : Microcosm Press.

[28] Peter M. Shames, Marc A. Sarrel (2015) *A Modeling Pattern for Layered System Interfaces*, 25th Annual INCOSE International Symposium: INCOSE.