KiboCUBE Academy

Lecture 10 – Second Edition

Introduction to CubeSat Command and Data Handing System

Tohoku University

Department of Aerospace Engineering

Associate Professor Dr. –Ing. Toshinori Kuwahara

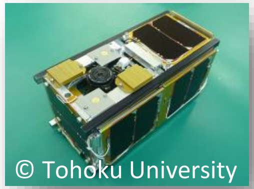
This lecture is NOT specifically about KiboCUBE and covers GENERAL engineering topics of space development and utilization for CubeSats.
The specific information and requirements for applying to KiboCUBE can be found at:
https://www.unoosa.org/oosa/en/ourwork/psa/hsti/kibocube.html

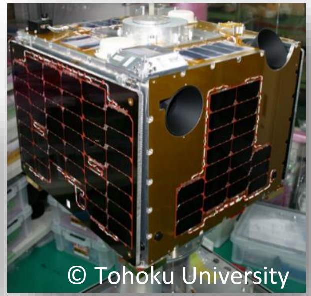
JAXA

UNISEC JAPAN
University Space Engineering Consortium

United Nations Office for Outer Space Affairs ★ Japan Aerospace Exploration Agency
Kibo CUBE

Lecturer Introduction





Toshinori Kuwahara, Dr. –Ing.

Position:

2015 - Associate Professor, Department of Aerospace Engineering, Tohoku University
2017 - Technical Advisor, Nakashimada Engineering Works, Ltd.
2017 - Technical Advisor, ALE Co., Ltd.
2020 - Chairperson, University Space Engineering Consortium Japan (UNISEC)
2021 - Co-founder/CTO, ElevationSpace Inc.

Research Topics:
Space Development, Utilization, and Exploration by Small Spacecraft Technologies

Contents

1. Introduction to Command and Data Handling System
2. CubeSat C&DH Hardware Components
3. CubeSat C&DH Software
4. Space Environmental Effects on C&DH System
5. CubeSat System Integration
6. Functional Verification of C&DH System
7. Conclusion

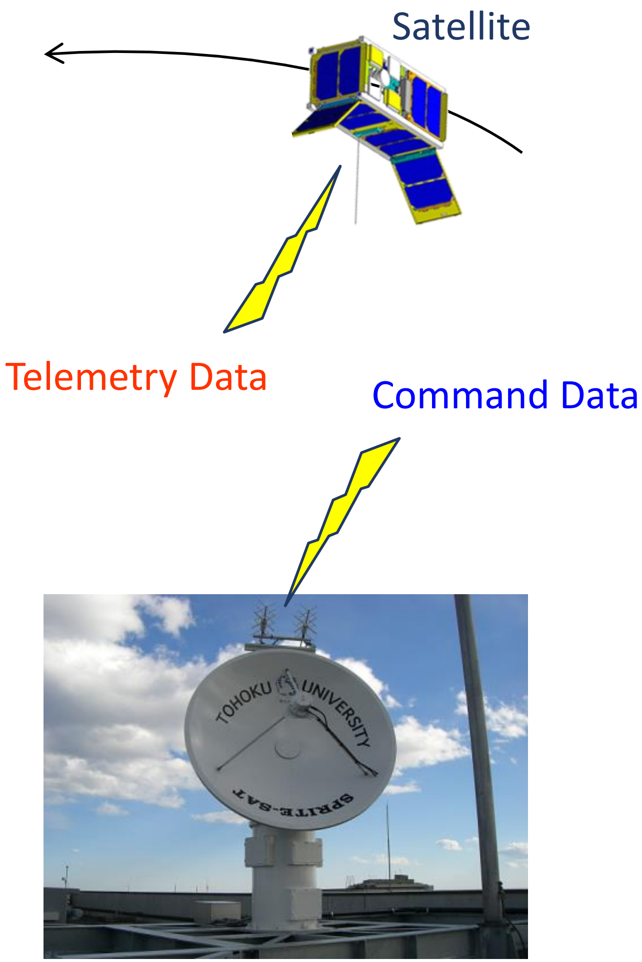
1. Introduction to Command and Data Handling System

1. Introduction to Command and Data Handling System

1.1. Functionalities of Command and Data Handling System

The Command and Data Handling System (C&DH) is one of the satellite subsystems, which are responsible for the processing of commands, handling of on-board data, and to some degree, execution of on-board autonomous functions.

- **Command Data** – Data sent from ground stations to satellites in order to control the satellite system. A single command is usually arranged as a relatively small data packet.
- **Telemetry Data** – Data sent from the satellite to ground stations.
    - **Housekeeping (HK) Data** : Basic status information of satellite components in order to monitor the housekeeping status of the satellite. HK data is usually periodically and continuously sent to the ground stations during contact.
    - **Mission Data**: Data acquired from the mission instruments, such as Earth observation images, measured environmental sensor data, received communication signals, etc. The amount of mission data is usually very large and a long communication period and/or high-speed communications are required.

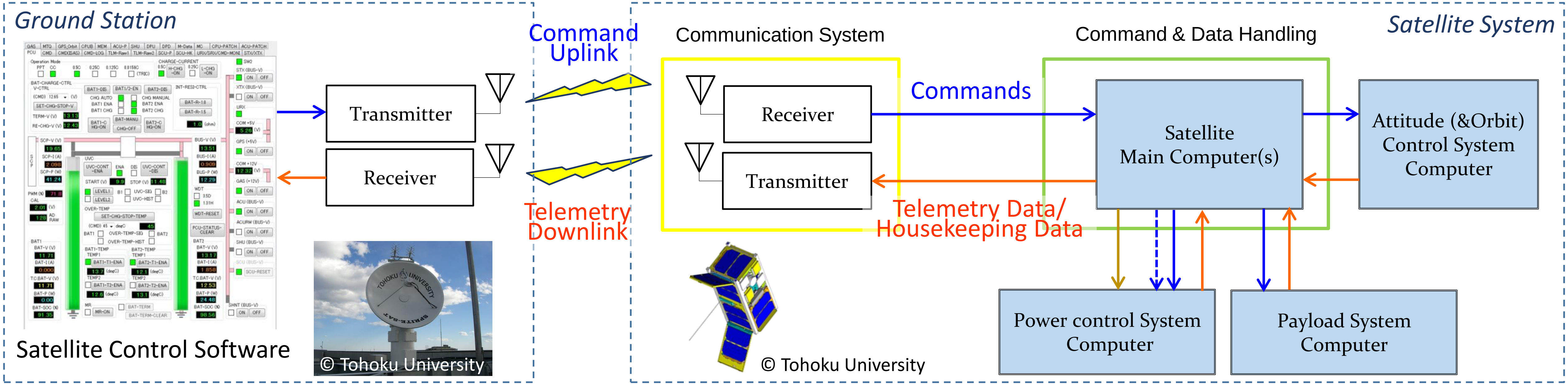


Ground Station

1. Introduction to Command and Data Handling System

1.2. Scope of Satellite Commanding and Status Monitoring

- The C&DH subsystem conducts command processing, on-board data handling, and autonomous functions.



Command Processing:

- Command decoding and validation
- Command processing and distribution
  - Command data packet →
  - Discrete pulses - - - - - →
- Decryption

Data Handling:

- Housekeeping (HK) data collection
- HK data processing and formatting
- Mission data acquisition and formatting
- Downlink data formatting and coding
- Encryption

Autonomous Functions:

- Contingency situation detection (e.g. UVC: Under voltage control)
- Transition to safe mode (Automatic power-off of components)

1. Introduction to Command and Data Handling System

1.3. Satellite Commanding

There are several classifications for the types of commands.

Execution Timing:

- Real time command – A command which is executed right after it is received by the on-board computer.
- Stored command – A command which is once stored in the memory of the on-board computer and executed at the specified execution time based on the on-board real time clock (RTC). Also known as a "Time-tagged command" or a "Time-stamped command".

Data Structure:

- Discrete Command – A command which is executed as a single command data packet.
- Block command – A set of more than one command data packets to be upload without packet loss, in order to uplink a large amount of stored commands, software update packages, calibration tables for sensors, etc.

Command Packet

Header	Command Contents	Footer

Block Command

Header	Command Contents	Footer
Header	Command Contents	Footer
	⋮	
Header	Command Contents	Footer

1. Introduction to Command and Data Handling System

1.4. Command Execution – Stored Command

For the secure execution of stored commands, satellites need to be equipped with time information, and the command format needs to include execution time and/or time interval information.

On-board Real Time Clock (RTC):

- A real time clock is usually implemented into the satellite main on-board computer in a format, which is intuitively understandable by the satellite operator, such as UTC (Coordinated Universal Time).
- UTC information can be obtained by GPS receivers and the RTC of the on-board computer can be synchronized to the information. Together with the PPS (Pulse Per Second) output from the GPS receiver, precise timing can be achieved even in Low Earth Orbit.
- UTC information can also be uploaded from the ground station for satellites without GPS receivers or as a backup operation. However, the timing precision is low due to the communication latency and limitation in accuracy of the on-board oscillator frequency.

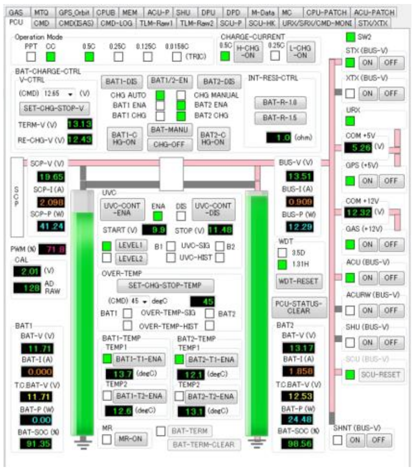
Command Format:

- The command format shall support on-board RTC synchronization function.
- Absolute command execution time shall be set through commands, such as YYYY/MM/DD/HH/mm/SS, etc.
- It is convenient if the command execution interval relative to the previously executed command can be set.

1. Introduction to Command and Data Handling System

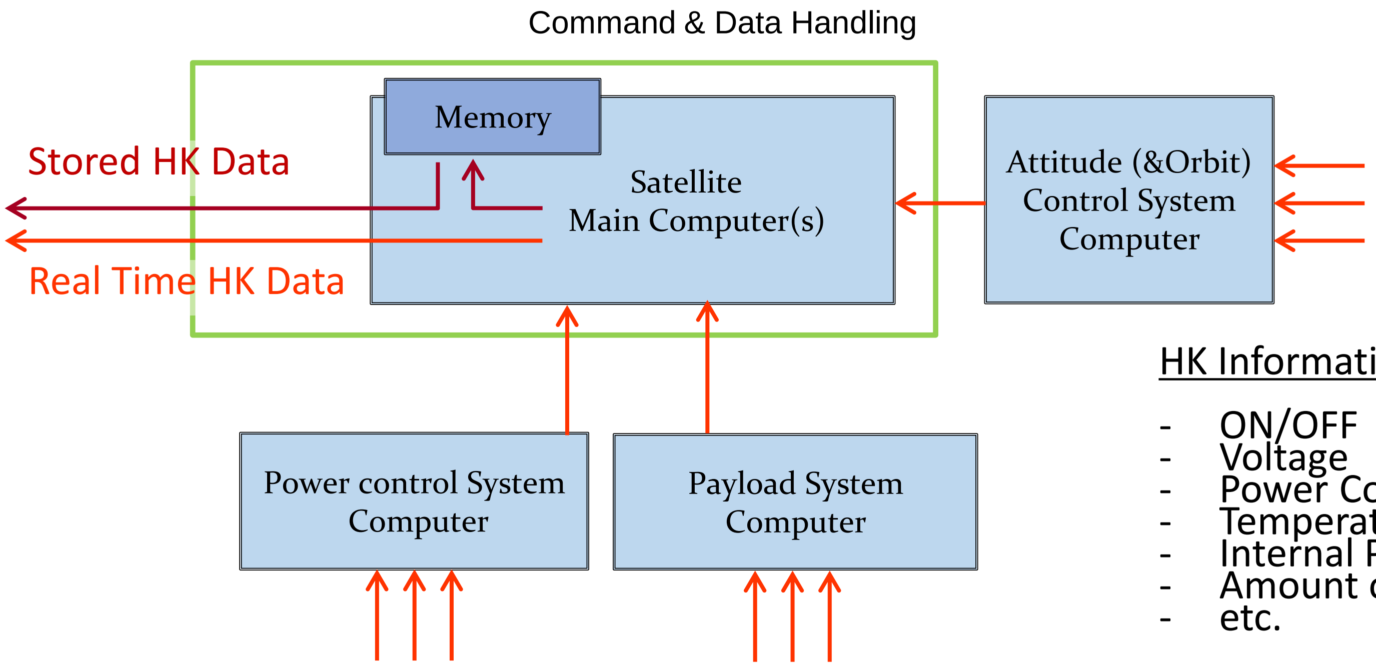
1.5. Satellite Monitoring

- The most common way of implementing satellite monitoring functionality is that the all the satellite components that are powered on deliver their housekeeping data periodically and continuously to upstream devices.
- The main computer receives, processes, and formats these HK data and stores it to the on-board memory, often configured in a ring-buffer manner.
- When ground contact is available, the on-board computer sends the real time HK data and stored HK data to the communication system for the downlink.



Satellite Control Software

© Tohoku University



HK Information

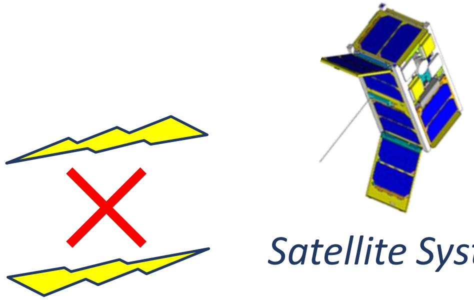
- ON/OFF
- Voltage
- Power Consumption
- Temperature
- Internal Parameter
- Amount of Mission Data
- etc.

1. Introduction to Command and Data Handling System

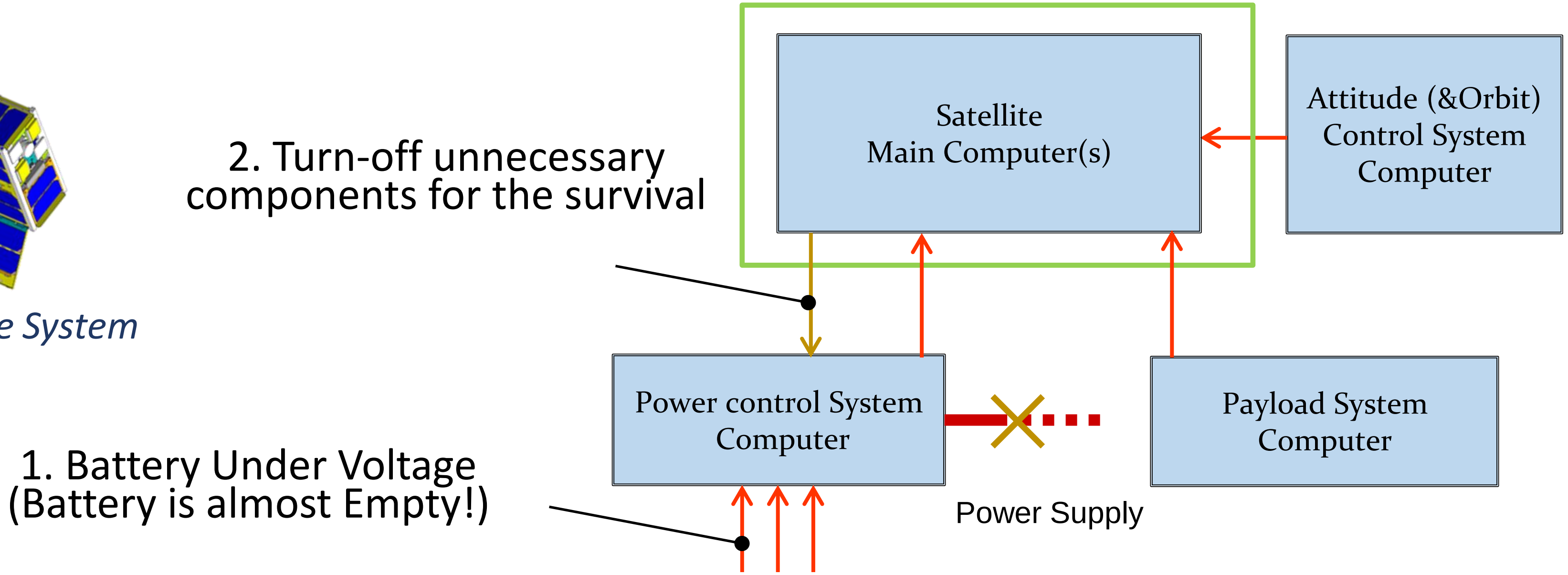
1.6. Autonomous Functions

- Satellites need to be equipped with a certain degree of autonomous functions.
- Power budget monitoring and management is especially one of the most important autonomous functions.
- If the satellite detects a battery under voltage, which is indicating that the battery is nearly depleted, the satellite autonomous function shall eliminate the excess power consumption by securely turning off unnecessary components to move into a safe mode for survival, and wait until the battery state-of-charge (SOC) recovers to a sufficient level.

Ground Station



Satellite System

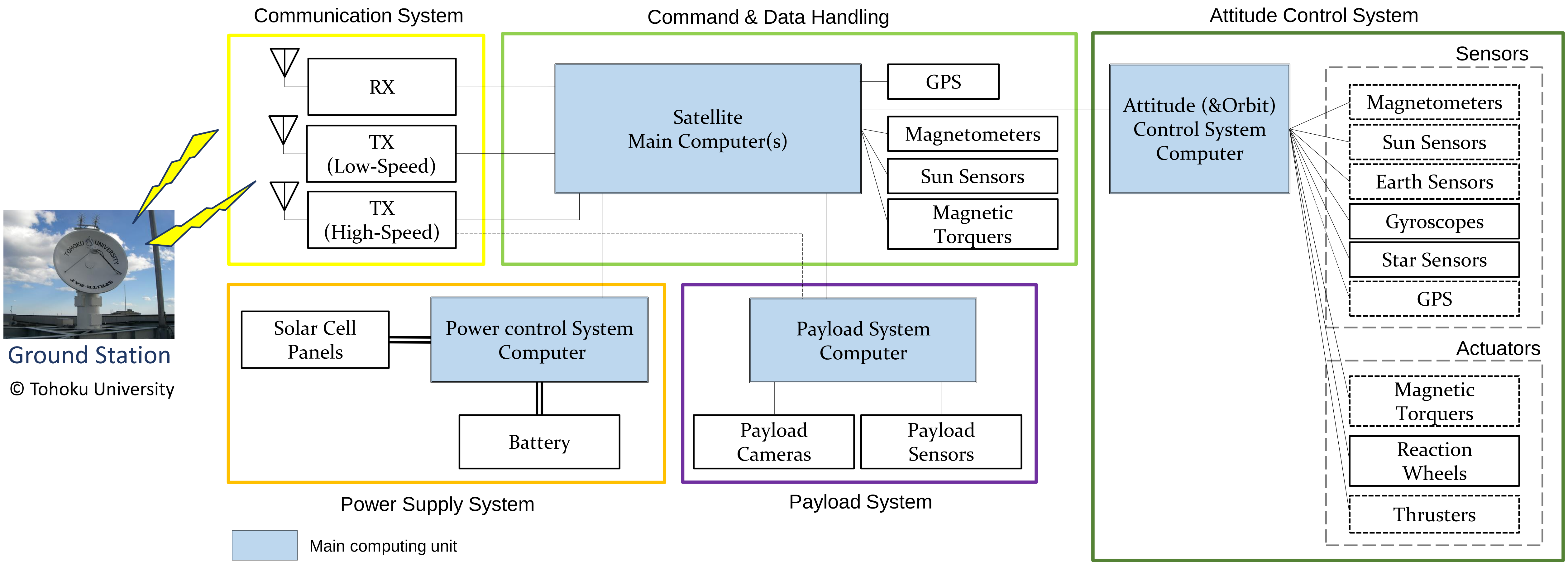


2. CubeSat C&DH Hardware Components

2. CubeSat C&DH Hardware Components

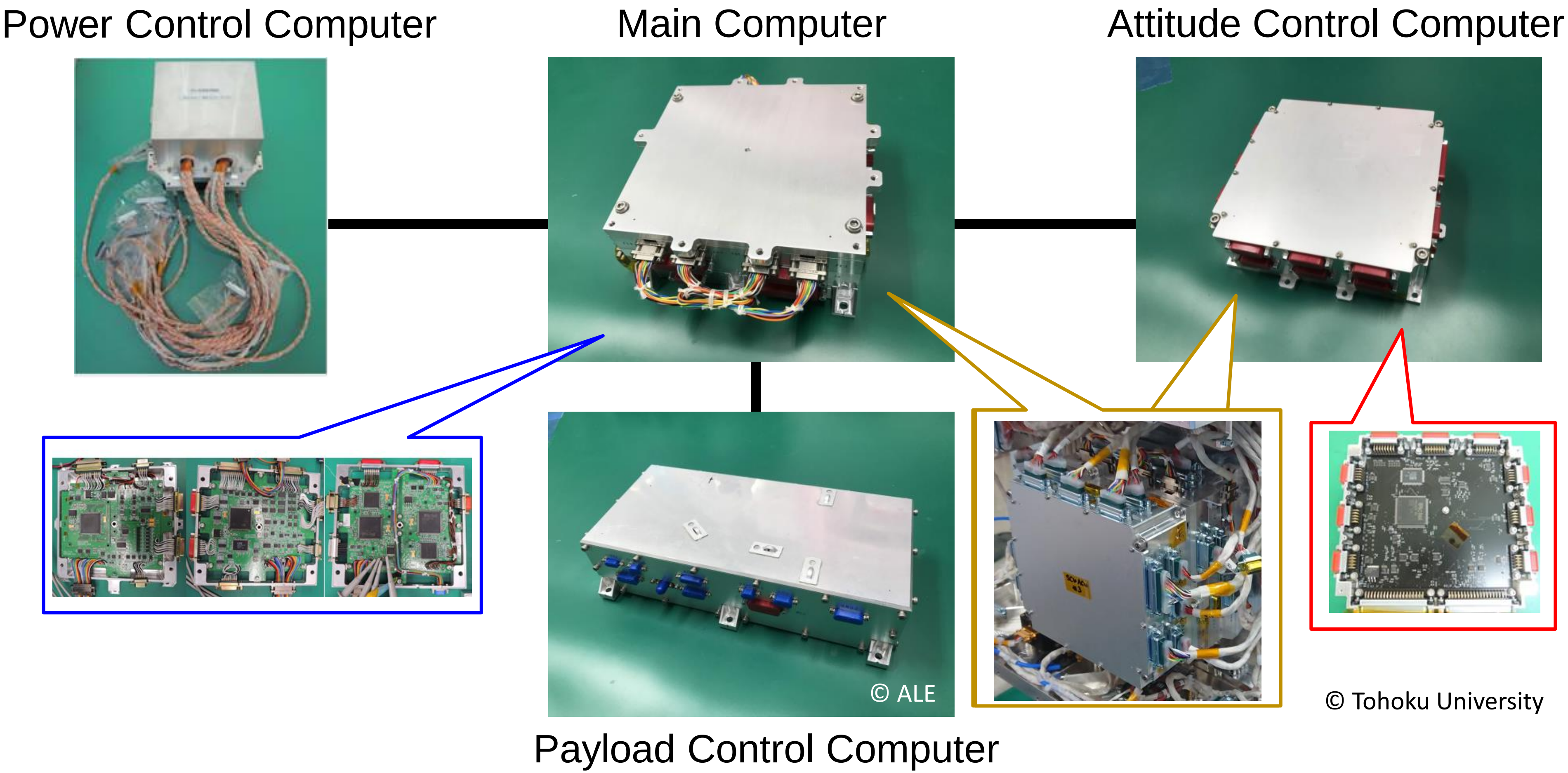
2.1. Satellite System Block Diagram

- System architecture can be described using a system block diagram. Satellite components can be generally categorized into subsystems according to their functionalities.
- Each interface needs to be specified, controlled, integrated, and tested for the system integration.



2.2. Satellite System Computers

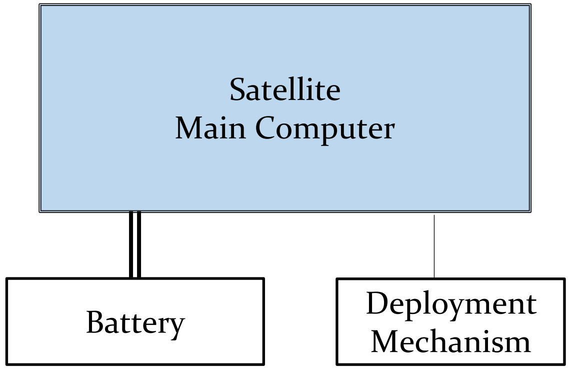
Example of On-board Computers (Micro-satellite RISESAT)

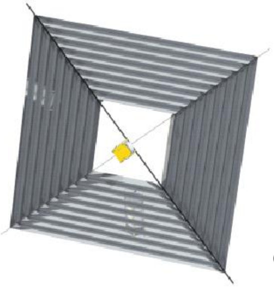


2. CubeSat C&DH Hardware Components

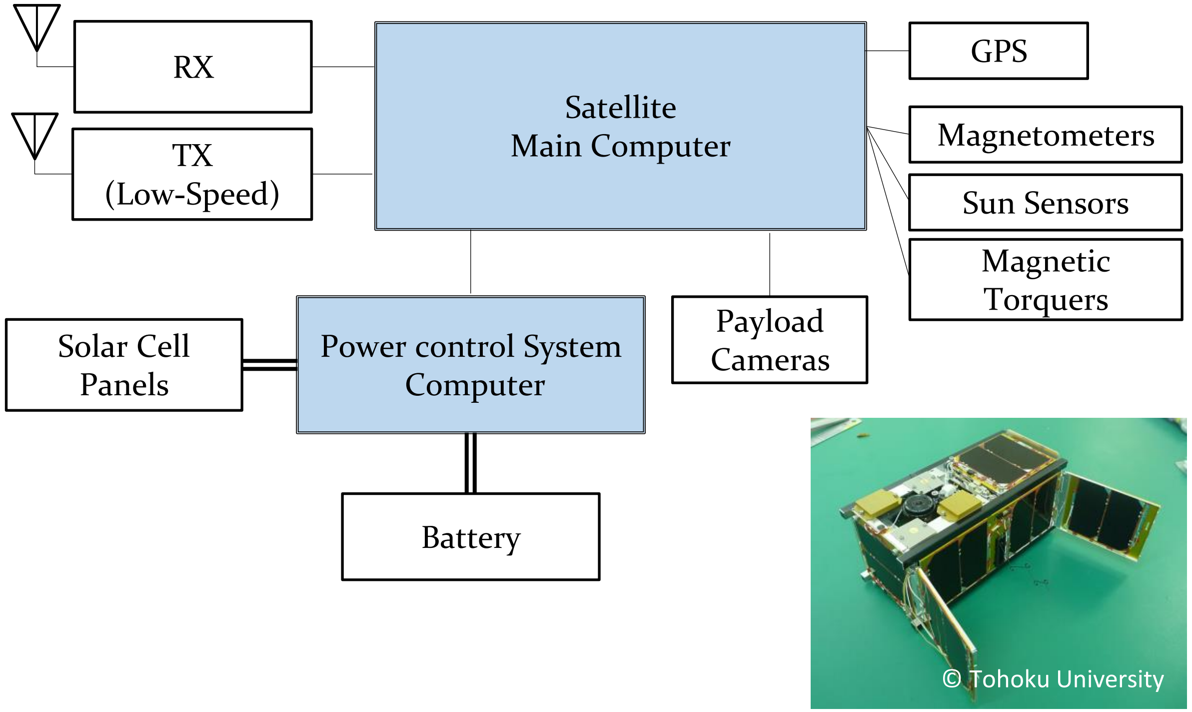
2.3. Electrical Configuration Design

- Satellite system architecture reflects the complexity of the system, and depends on the mission objectives and requirements.
- Fulfilling the mission requirements with minimum component configuration is important to achieve a high system reliability.





Implementation example of 1U CubeSat FREEDOM



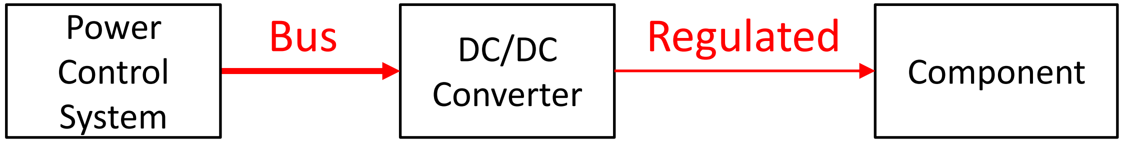
CubeSat system with payload camera and coarse attitude control system

2.4. Electrical Interface

Types of electrical interface can be classified as follows:

Power supply interface

- Unregulated/Bus: Satellite bus voltage, which fluctuates depending on the state of charge (SOC) of the battery.
- Regulated: Voltages-regulated power supply through DC/DC converters, such as 3.3V, 5V, 12, 28V, etc.

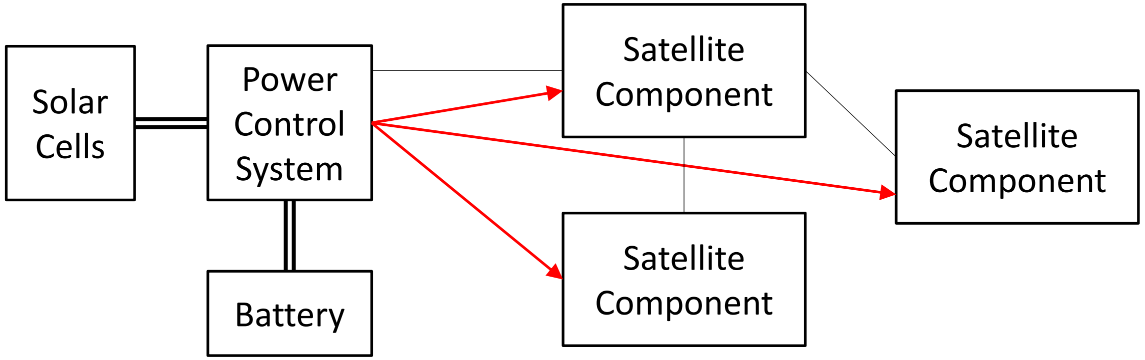


Signal interface

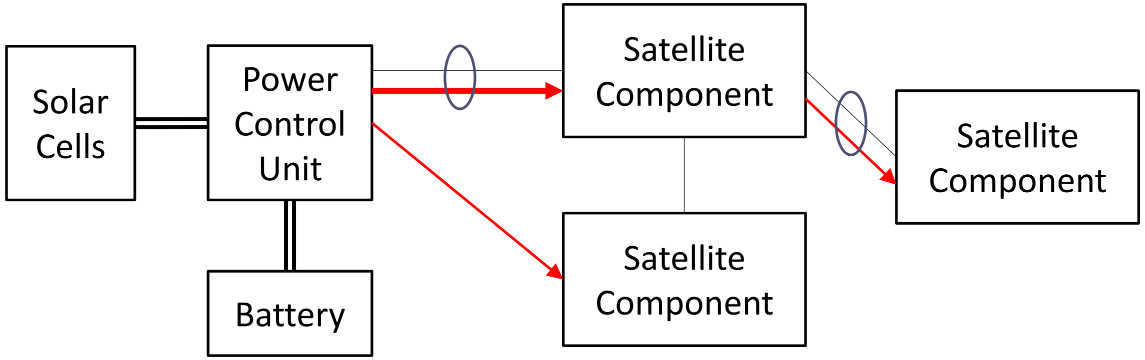
- Analog signal
  - Active analog: voltage output from analog sensors, voltage output from powered devices.
  - Passive analog: thermistors (power is supplied from outside to measure the value)
- Digital signal
  - Discrete signal: ON/OFF status of mechanical switches, status of electrical circuits, etc.
  - Synchronous Serial: Communication interface with dedicated clock signals, suitable for high-speed communication.
  - Asynchronous Serial: Communication interface without dedicated clock signals, suitable for low-speed communication with less cables.
- Radio Frequency
  - Communication between satellite and ground station. Require international radio frequency coordination (before launch), and license.
  - Communication between satellites, between components inside a satellite, etc.

2.5. Power Distribution Method

- Power distribution from the power control system to satellite components can be direct or in-direct, mainly depending on the configuration of the mechanical integration of the satellite electrical components.
- Harness system becomes simpler when the power supply lines and signal lines can be combined into a harness assembly between two devices.
- For a relatively bigger satellite system, hub configuration becomes more efficient.
- The voltage of unregulated power supply lines is usually higher than the regulated voltage, and hence, more efficient in power distribution over a long harness.
- Attention needs to be paid that a long harness causes a considerable voltage drop, and also that a high current flow generates noticeable magnetic fields which causes disturbance to the satellite attitude stability.



Point-to-Point Star Configuration

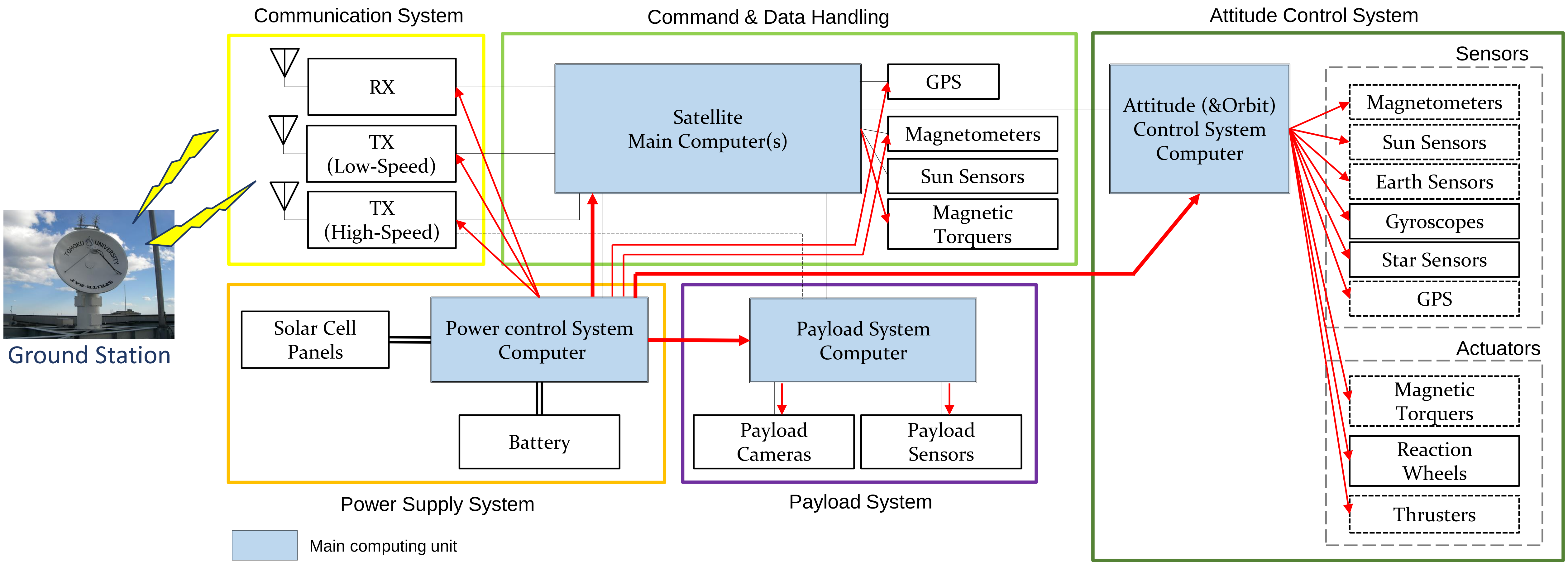


Hub Configuration

2. CubeSat C&DH Hardware Components

2.6. Power Distribution Architecture

- Power distribution architecture shall be carefully designed especially for small space systems in order to reduce the system complexity, number of components, harness, mass, size, volume, and power inefficiency.



2. CubeSat C&DH Hardware Components

2.7. Communication Interface

- There are several types of signal interfaces commonly applied to satellite systems.

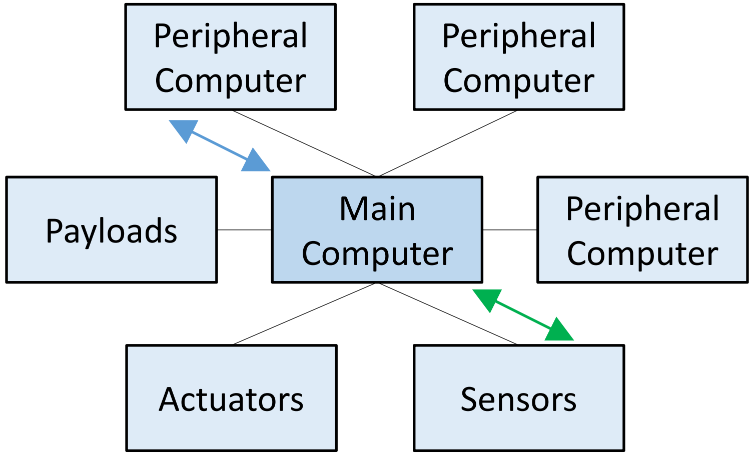
Interface Name	Signal Lines	Clock	Topology	Protocol
RS-232C	Single End	Asynchronous	Point-to-Point	UART
RS-422	Differential	Asynchronous	Point-to-Point	UART
RS-485	Differential	Asynchronous	BUS	UART
SPI	Single End	Synchronous	BUS	SPI
I^2C	Single End	Synchronous	BUS	I^2C
USB	Differential	Asynchronous	BUS	USB
SpaceWire	Differential	Asynchronous	Point-to-Point	SpaceWire
Ethernet	Differential	Asynchronous	Point-to-Point	Ethernet

Physical Layer	Voltage
TTL (Transistor-Transistor Logic)	3.3V / 5V
LVDS (Low Voltage Differential Signal)	3.3V

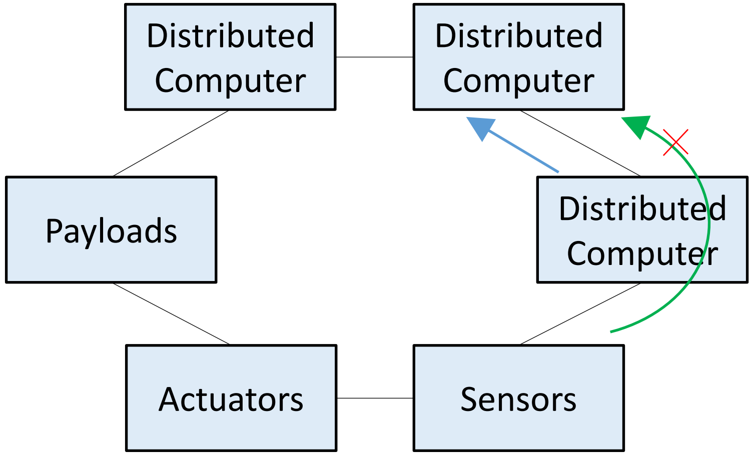
2. CubeSat C&DH Hardware Components

2.8. Computer Network Architecture

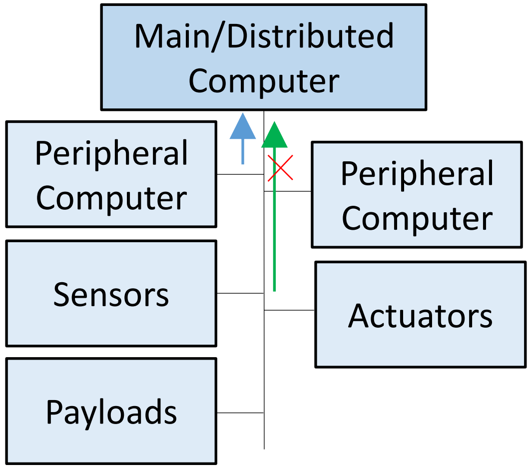
- The computer network architecture shall be carefully designed for developing a computer system for satellites.
- There are several different computer network architectures, as below:



Centralized (Star) Architecture



Ring Architecture

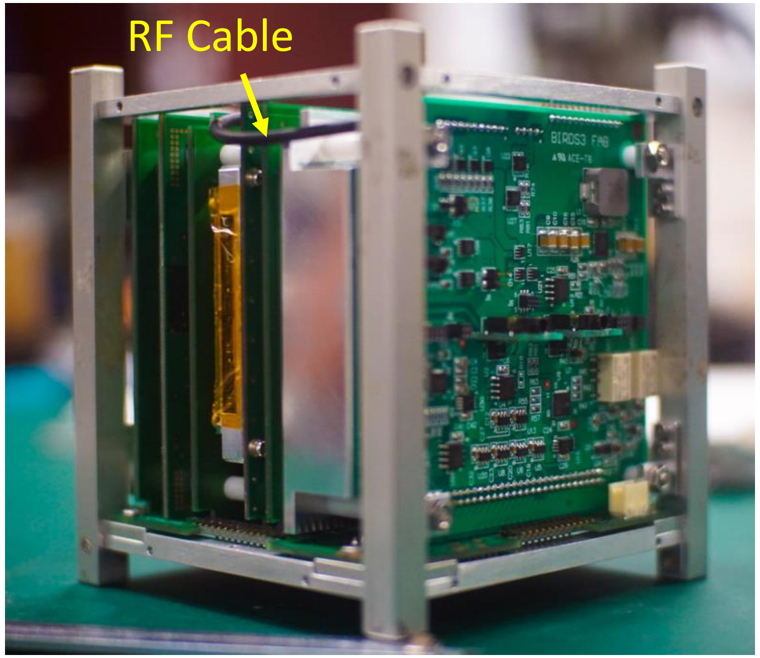


Bus Architecture

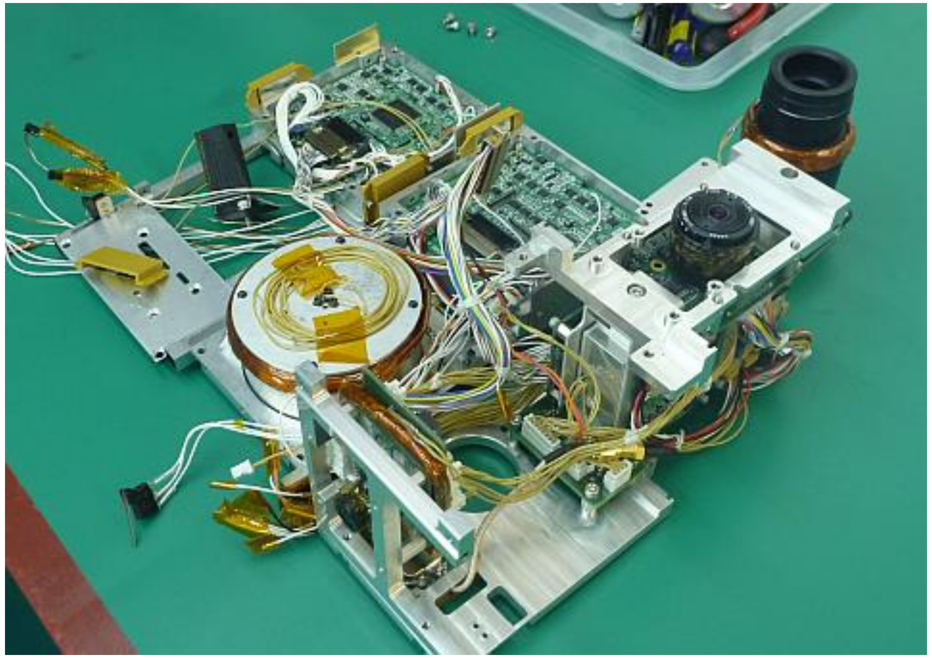
Architecture	Centralized (Star) Architecture	Ring Architecture	Bus Architecture
Merits	High integrity, direct interface only Less influence of unit failure	Smaller wiring harness Easy to add new node	Smaller wiring harness Easy to add new node
Demerits	Point-to-Point connection Large wiring harness	More influence of unit failure Low communication throughput	Interface needs to be compatible Low communication throughput

2.9. Radio Frequency interface

- Radio Frequency interface needs a special point-to-point harness connection.
- RF connectors are relatively big and RF cables are relatively thick due to the shielding materials.
- Point-to-Point connection requires harness wiring, which consumes space inside the satellite, and mass budget.
- Harness wiring needs special attention for the accessibility and integration procedure.



1U CubeSat © Kyutech

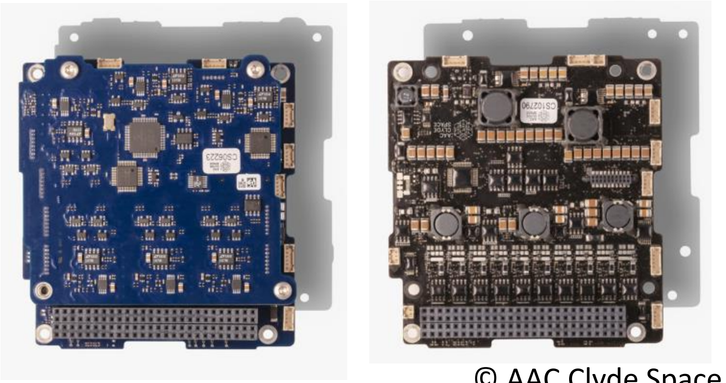


2U CubeSat RAIKO © Tohoku University

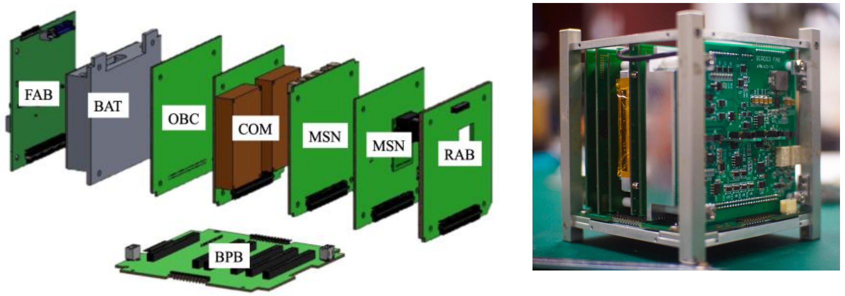
2. CubeSat C&DH Hardware Components

2.10. Mechanical and Electrical Interface Standardization

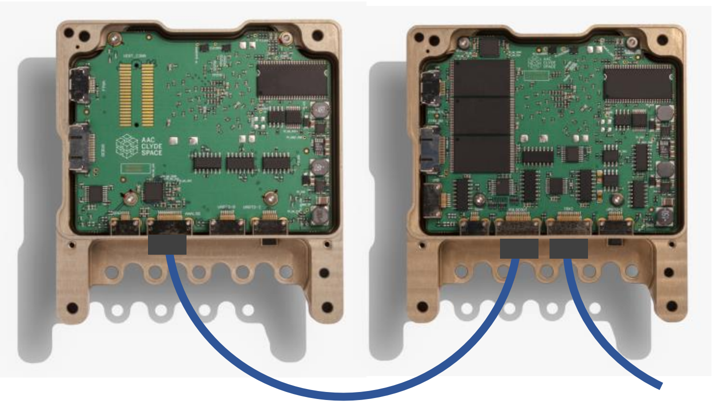
- Three major ways of mechanical and electrical integration: PC-104 Style, Backplane Style, and Point-to-Point.
- The scope of the standards include mechanical interfaces, connectors, types of signals, and pin assignments.



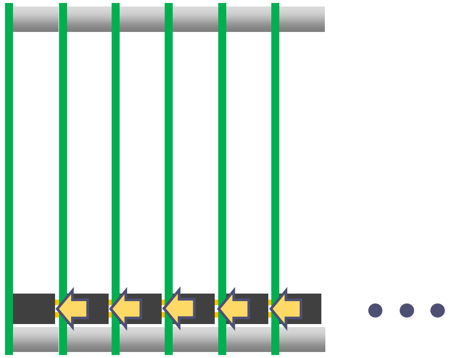
© AAC Clyde Space



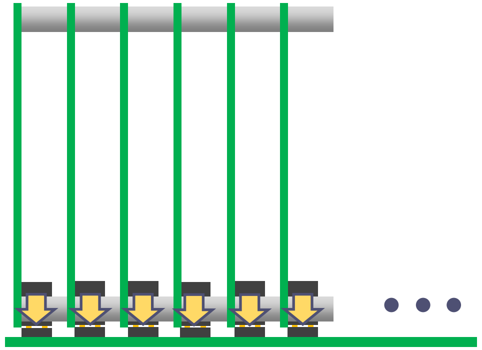
© Kyutech



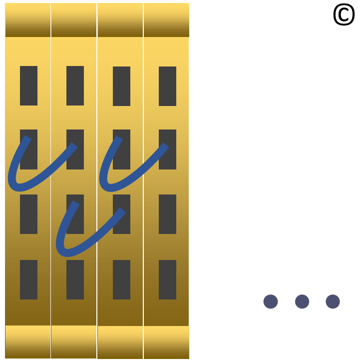
© AAC Clyde Space



PC-104 Style



Backplane Style



Point-to-Pont

2. CubeSat C&DH Hardware Components

2.11. CubeSat Subsystem Interface Definition – UNISEC-Europe

- Standardization of CubeSat specification can expedite world-wide application of CubeSats.
- CubeSat subsystem interface standards suggested by UNISEC-Europe defines the interface specifications of the electrical interfaces based on the experiences of UWE projects, with the goal to promote a generic satellite platform.
- The design has been optimized with respect to mass, size, and energy efficiency, while trying to maintain a modular and flexible architecture.
- The proposed bus supports robust and rapid development, integration and testing of the satellite as well as simple maintenance, extension, and replacement of subsystems.

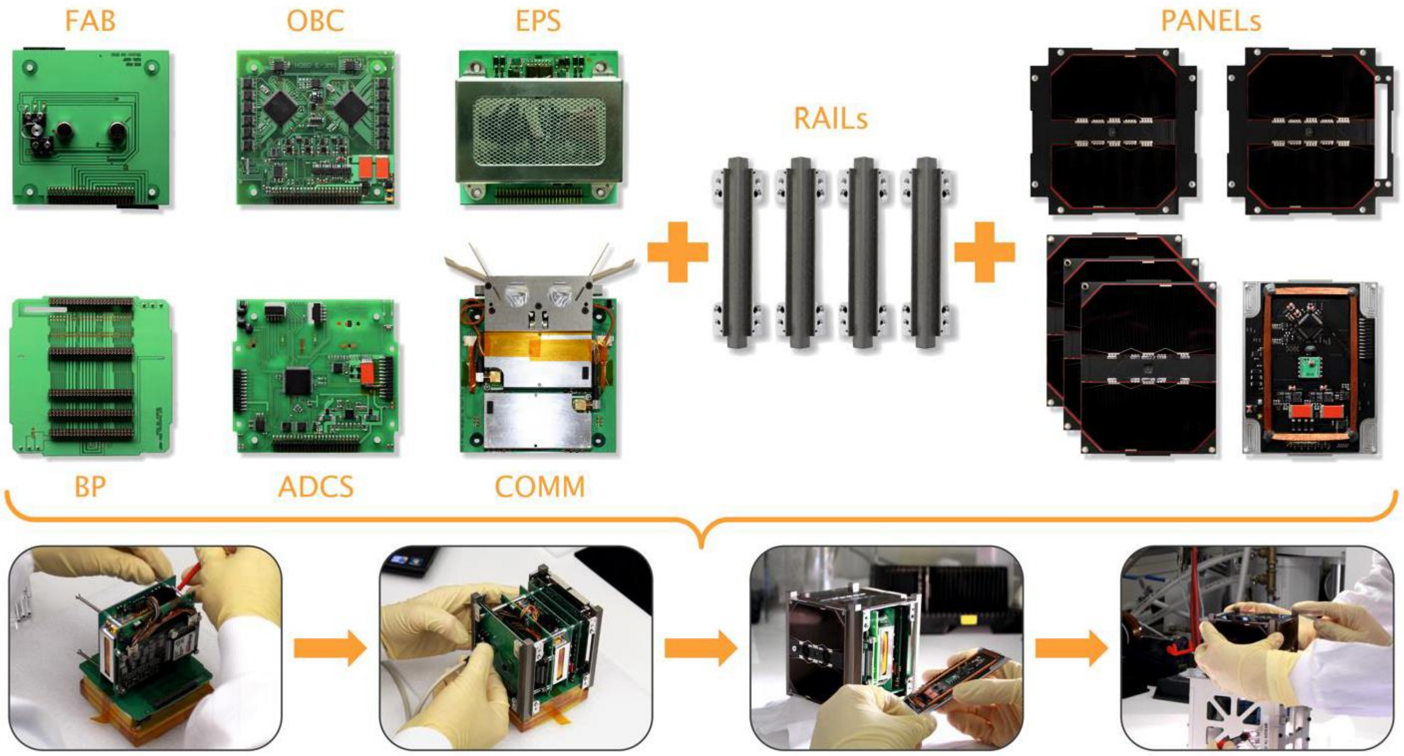


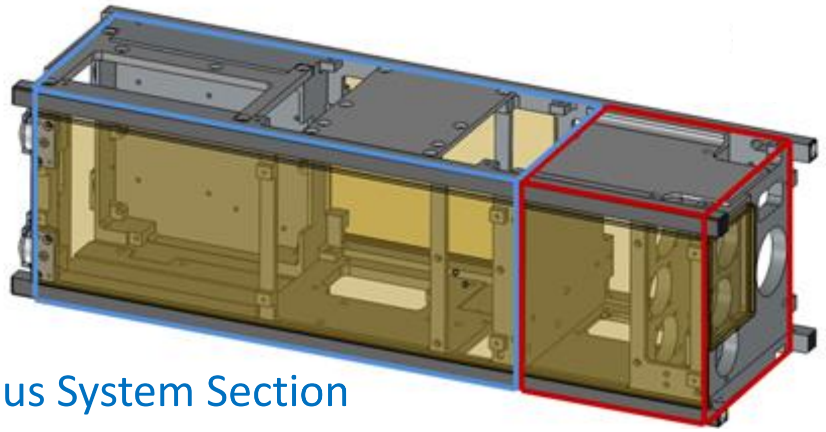
Figure: Overview of the modular UWE-3 pico-satellite bus being optimized for rapid integration and testing. Acts as a first reference implementation of the UNISEC bus.

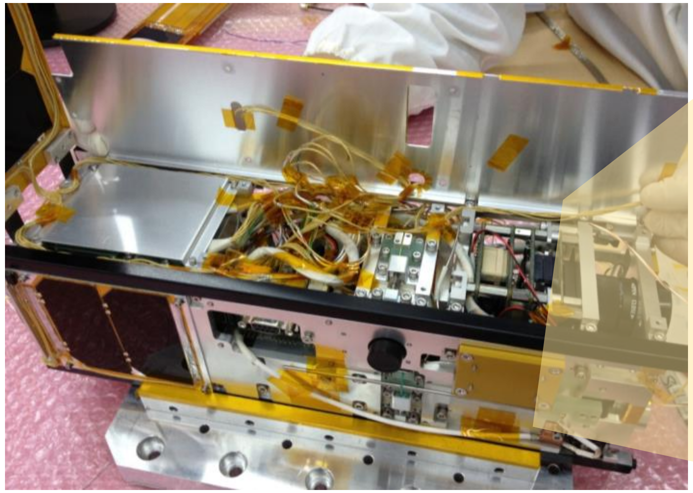
CubeSat Subsystem Interface Definition version 1.0

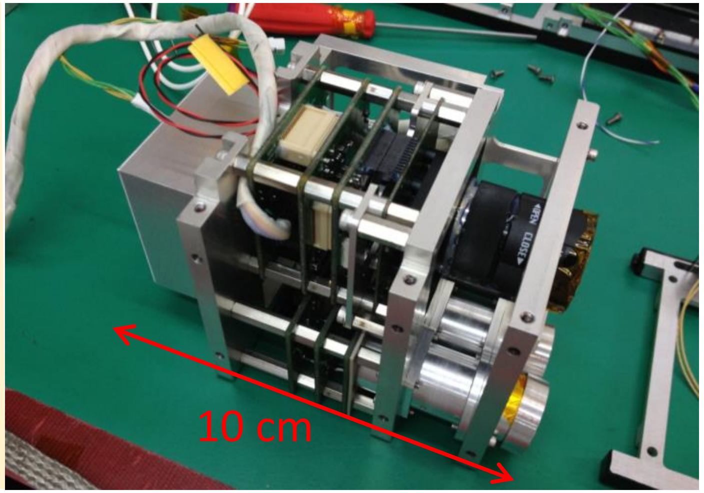
http://unisec-europe.eu/wordpress/wp-content/uploads/CubeSat-Subsystem-Interface-Standard-V2.0.pdf

2.12. Payload Interface

- Payload instruments usually tend to have custom interfaces with a combination of different types of electrical interfaces.
- Due to the limitation of mechanical envelope and maintainability, payload instruments need to be assembled as a unit by defining the mechanical, electrical, and thermal interfaces very clearly.
- Payload instruments related with radio frequency measurements are equipped with large antennas, which needs to be held down during the launch and deployed in orbit.







3U CubeSat S-CUBE © Chiba Institute of Technology / Tohoku University

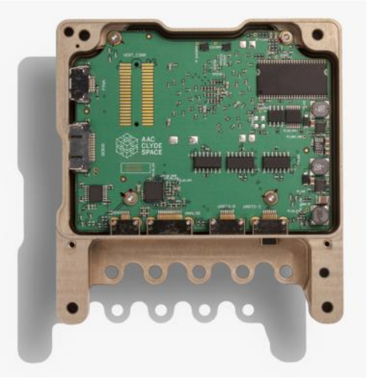
3. CubeSat C&DH Software

3.1. Requirements of C&DH Software

Satellite on-board software is responsible to achieve the required functions of the satellite with the available computer resources. Autonomous functions shall be implemented in software applications.

Characteristics of Computer System Performance

- Throughput (Processing speed, capability)
- Memory Size
- Number of interfaces and their communication speeds
- Number of available computer units
- Available/required electrical power and thermal stability
- Operating system, etc.



Example of CubeSat On-board Computer

Typical Requirements of Computer System

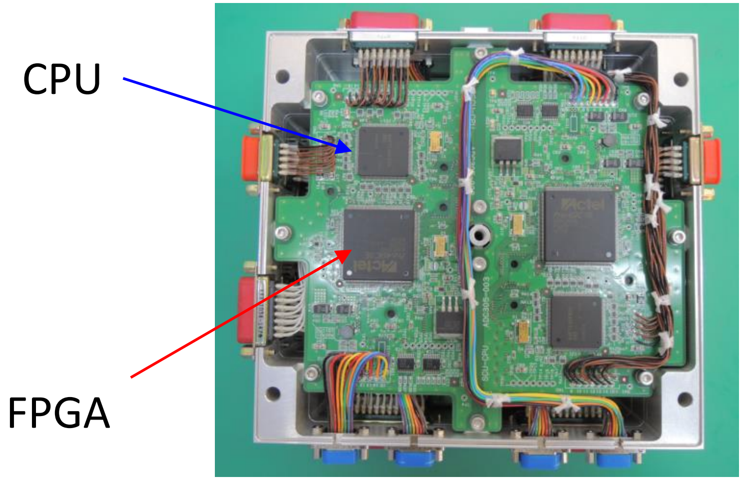
- Periodical processing of satellite control and status monitoring functionalities
- Secure and stable communication with ground station in real-time
- Management of timing, real-time clock
- Execution of stored command (Automated, Schedule-driven)
- Execution of autonomous functions, such as power management, thermal management, etc. (Autonomous, event-driven)
- Reliability, Maintainability, Testability, Reusability, etc.

3.2. Software and HDL (Hardware Description Language)

The most commonly used data processing IC devices for the C&DH computers are CPUs and FPGA. Combination of these devices can achieve integrated high-performance data processing capabilities.

Characteristics of IC devices

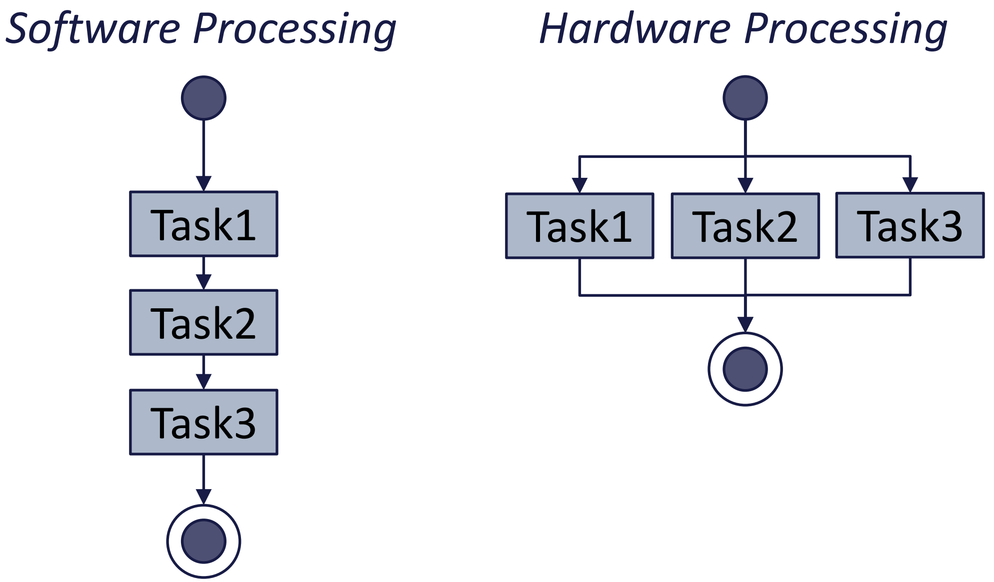
- **CPU: Central Processing Unit**
  - Based on Pre-embedded CPU architecture.
  - Execute software program.
  - Software instruction is executed one after another. (Series)
  - Programmed using software programming languages.
  - Flexible and functionalities can be expanded with larger program memories.
- **FPGA: Field Programmable Gate Array**
  - Programmable semiconductor logic devices.
  - Execute signal processing.
  - Signals are processed both in parallel and series.
  - Internal RTL (register transfer level) logic is designed using Hardware Description Languages (HDL).
  - Special-purpose-oriented and complexity of the functionalities are limited to the gate size of the hardware.



Example of On-board Computer

3.3. Programming Languages

There are different kinds of programming languages available both for software programming and hardware programming.



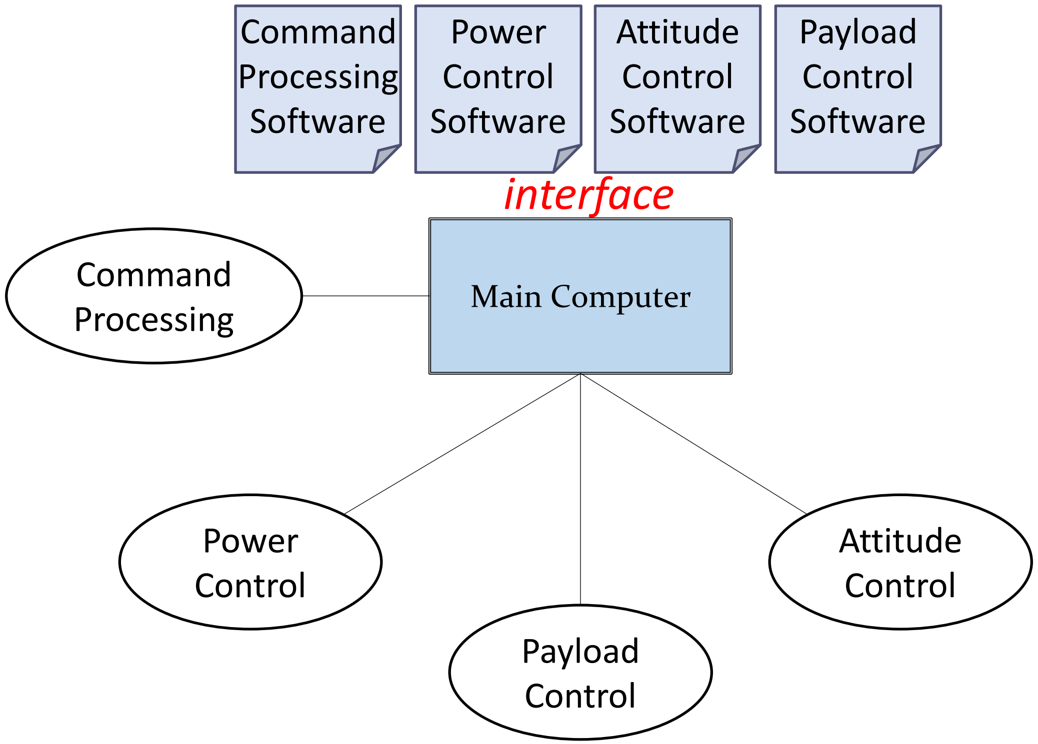
<u>Programing Languages</u>

- Software Programming of embedded systems
  - C
  - C++
  - Assembly language
  - etc.

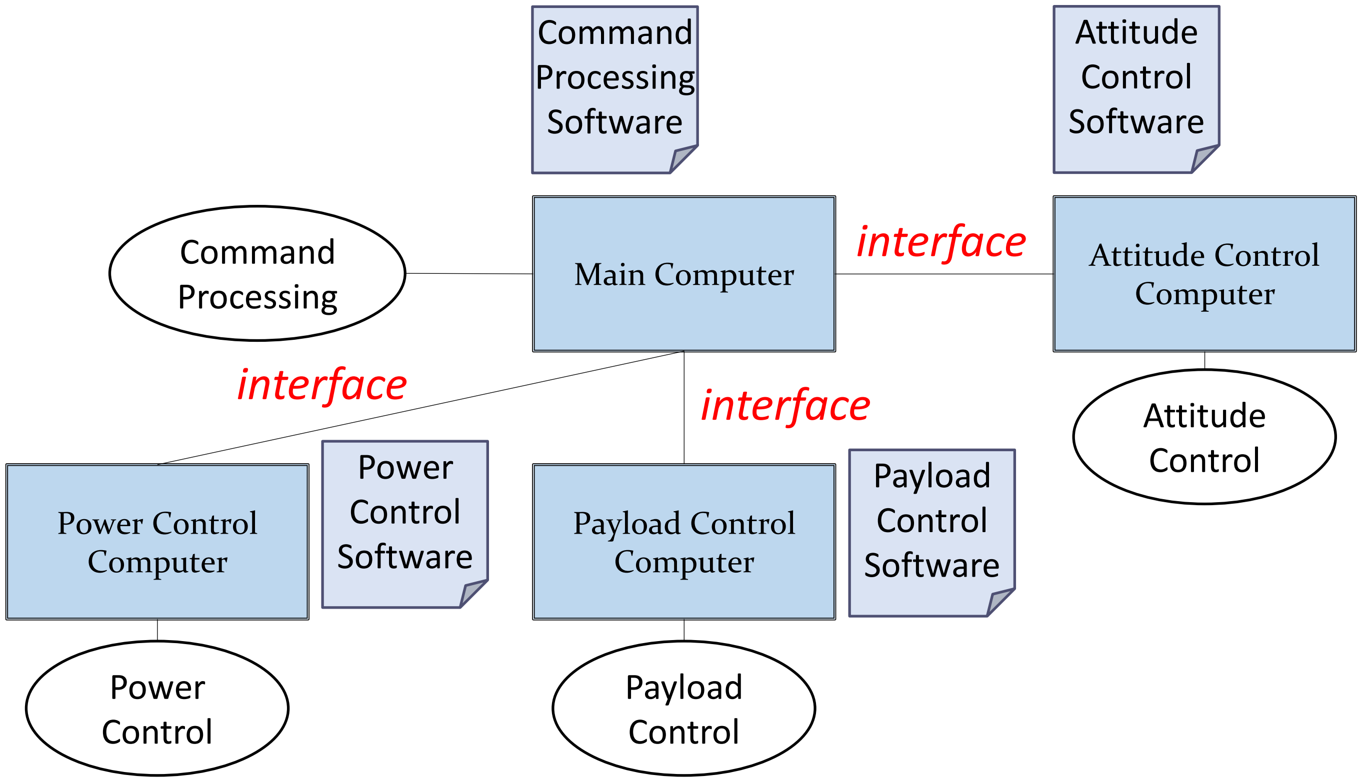
- Hardware Programming
  - VHDL - Very High-Speed Integrated Circuits (VHSIC) Hardware Description Language
  - Verilog, SystemVerilog
  - SystemC
  - etc.

3.4. Computer Architecture and Software Configuration

- Single computer architecture can realize a high degree of system integration with minimum hardware configuration. However, software development of each functional block depends on each other and tends to take more time than the distributed architectures, which enables parallel and independent development of each functional block.



Centralized Single Computer



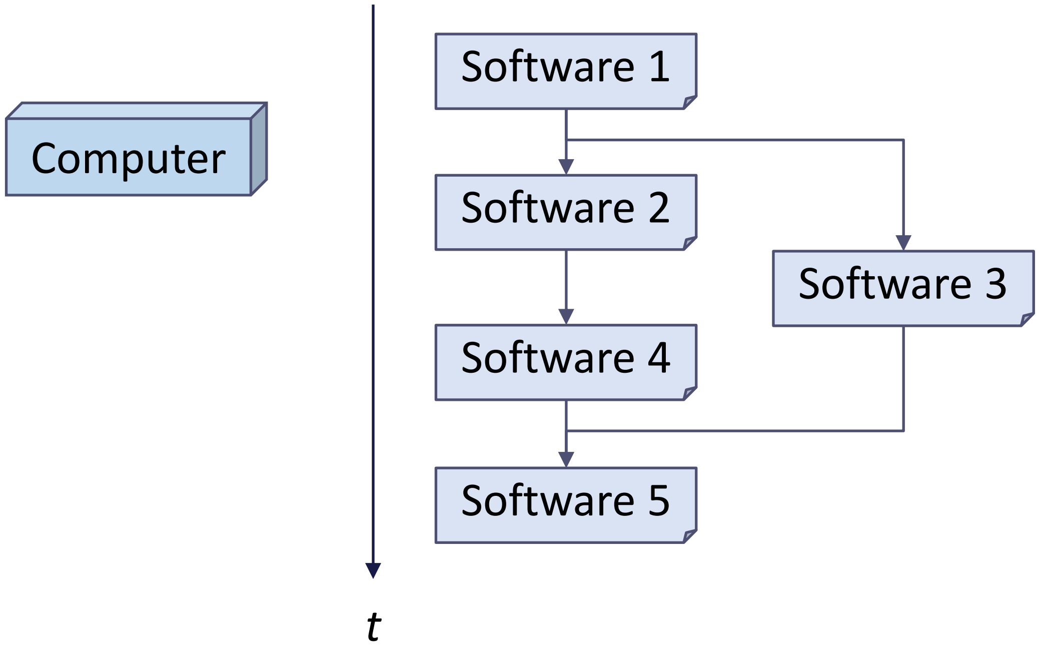
Distributed Computers

3. CubeSat C&DH Software

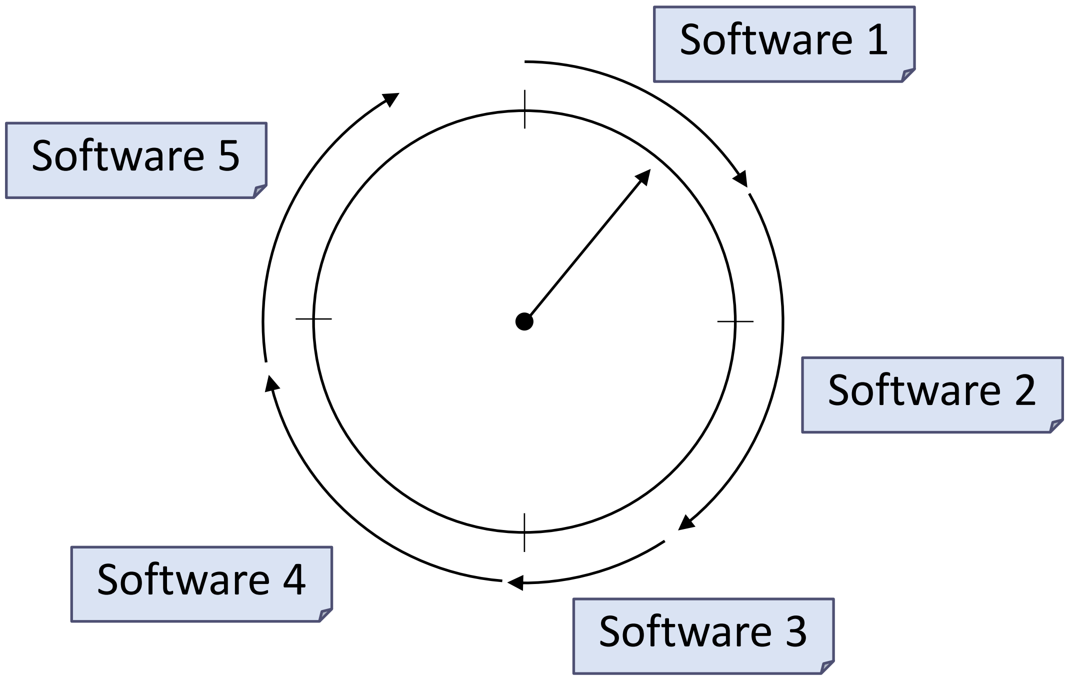
3.5. Software Integration

- Software units to be implemented and executed in a computer shall be integrated into a single piece of on-board software, paying attention to their execution order and logical relationships in between.
- The time period required for each of these software tasks shall be controlled so that the on-board computer can execute all the required tasks in the pre-defined required time period, periodically.

Software Execution



Periodical Execution

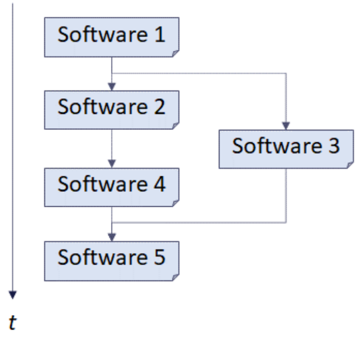


3. CubeSat C&DH Software

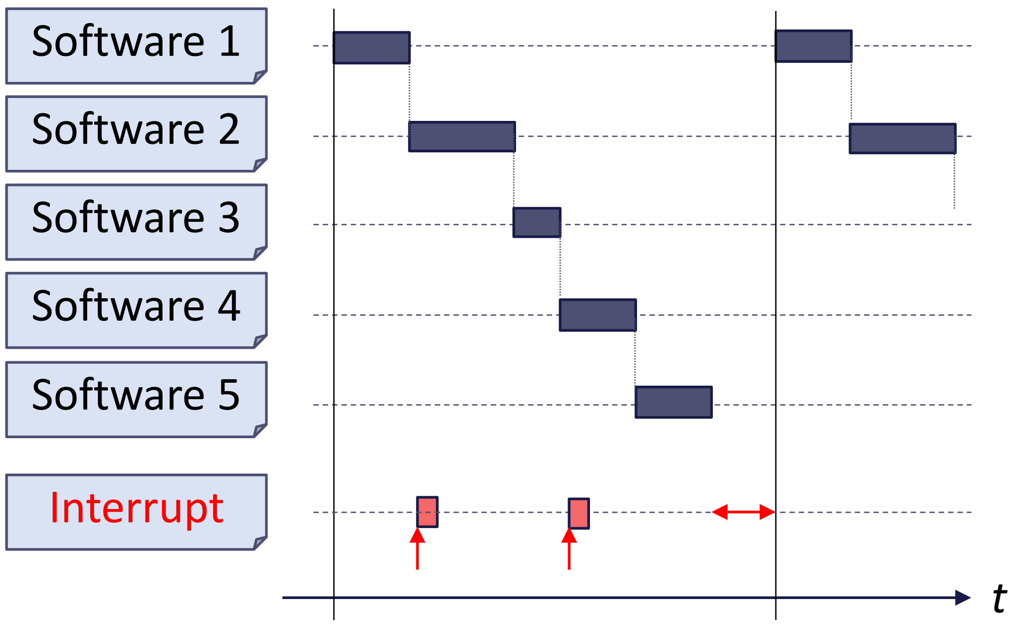
3.6. Software Operating System

- Software can be implemented either in a “bare-metal” way or using a Real-Time Operating System (RTOS).
- Relatively high-performance computers can utilize RTOS due to its overhead.
- Bare-metal implementation requires more precise tuning of the timing by the programmer.
- RTOS enables precise timing control of the execution and ease of parallel execution of more than one software units.

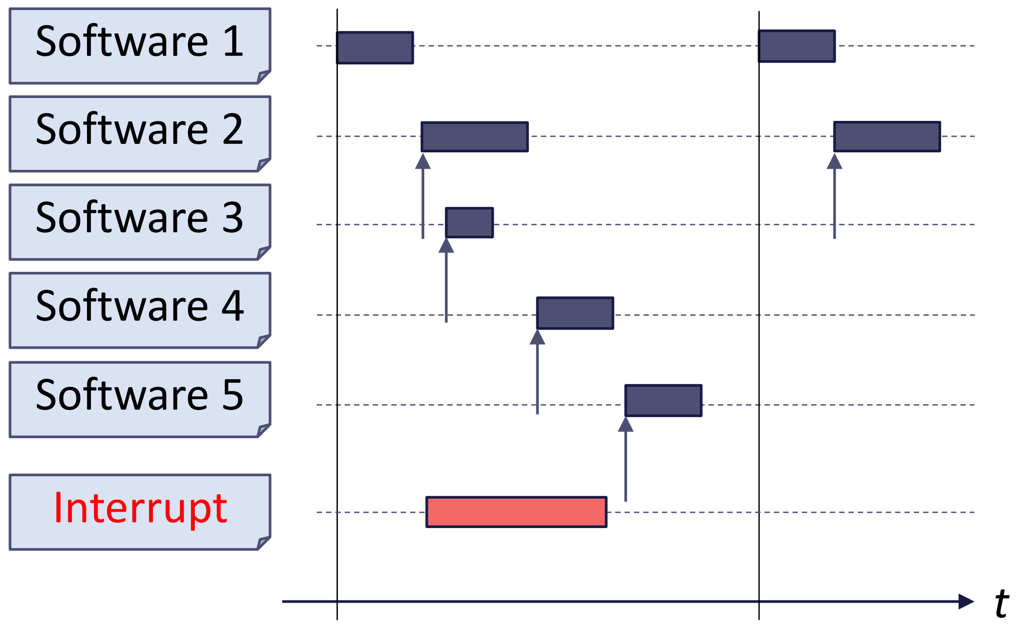
Software Execution



Bare-Metal Implementation



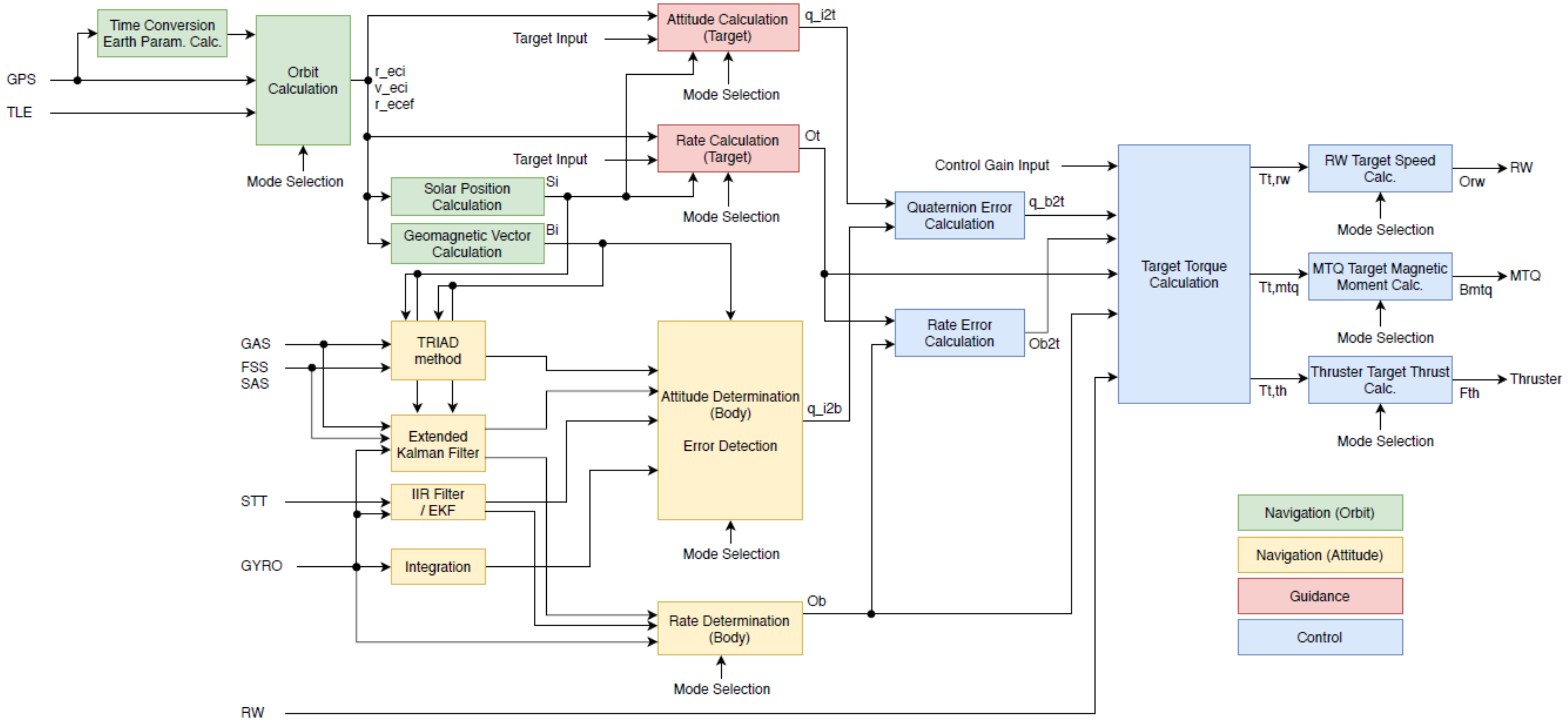
Real-Time Operating System Implementation



3.7. Attitude Determination and Control Software Implementation

- Attitude determination and control software processes includes software functional groups, such as guidance, navigation, and control, and their careful implementation to achieve high-precision attitude control.

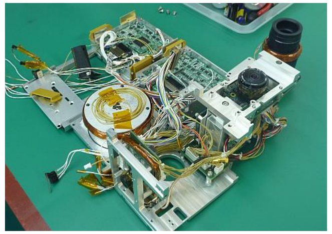
Attitude Determination and Control Software Process



3.8. Operational Planning

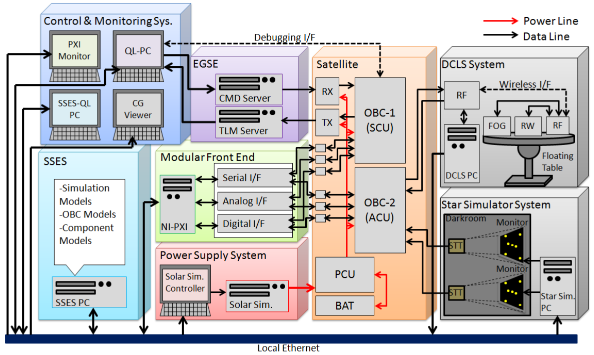
- Command list to be uploaded shall be verified beforehand, by means of ground verification before flight using the actual flight model or through ground verification using satellite simulator and satellite Engineering Model if available.
- A set of verified command sequence, i.e., the contents of the command, parameters, and time intervals, can be handled as a macro command, which can be further combined with other verified set of command lists.

Ground Verification before Flight



Verified Command List

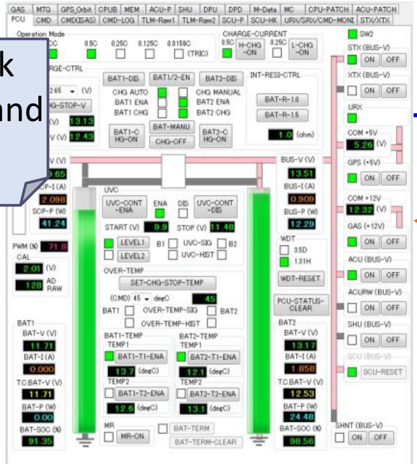
Ground Verification using Satellite Simulator



Verified Command List

Ground Station

Uplink Command List



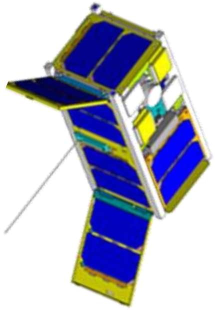
Satellite Control Software

Transmitter

Receiver

Command Uplink

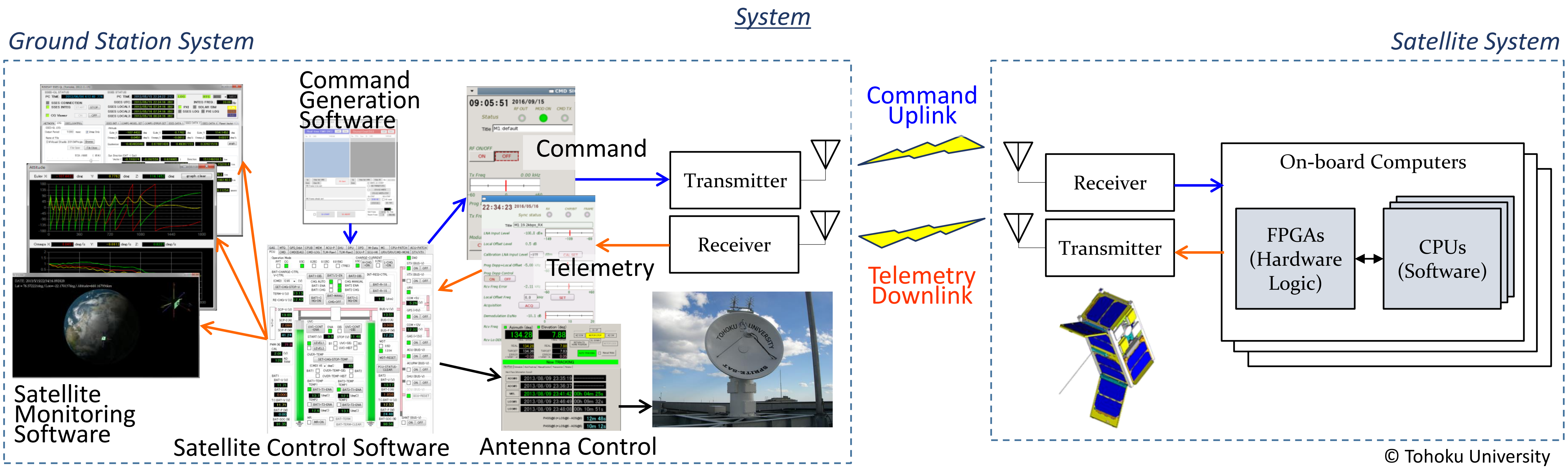
Telemetry Downlink



Satellite System

3.9. C&DH-related Software

- The software functionalities of both of the on-board computers and ground stations can consist of more than one software unit. It is important that the functional requirements and interface specifications are defined clearly.
- On-board software units: command execution, housekeeping data management, attitude control, etc.
- Ground station software units: command generation, housekeeping data analysis and visualization, ground station hardware control, etc.

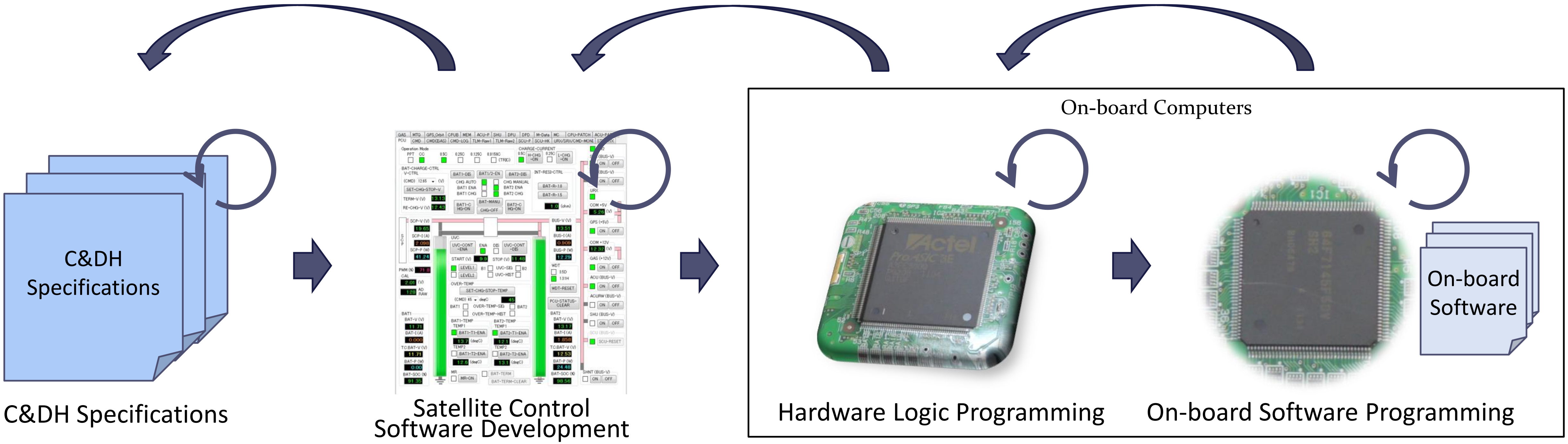


3. CubeSat C&DH Software

3.10. Development Process of C&DH Software

- The C&DH specification shall be defined first for the efficient development of related software programs.
- One of the recommended development approaches is to prepare satellite control software, and then hardware logic for FPGA devices, before programming and finalizing the detail of on-board software.

Iterative Design Process



4. Space Environmental Effects on C&DH System

4. Space Environmental Effects on C&DH Subsystem

4.1. Introduction to Space Environmental Effects on C&DH Subsystem

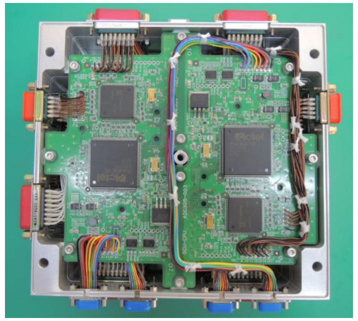
The C&DH system shall withstand following space environmental effects :

- Launch and Release Environment
  - Mechanical launch conditions: Vibration, Shock, Acoustic, etc.
    Harness system needs to be tightened securely!
  - Temperature, humidity, etc,

  * Small satellites, and hence C&DH subsystem, are usually powered off during the launch (Cold Launch).
- Vacuum
  - No cooling effects through convection expected, unlike ground applications.
  - The temperatures of integrated circuit chips tend to become quite high.
  - Active heat dissipation measures are required/recommended.
- Radiation
  - Electric circuits experience various kinds of radiation effects.
  - Radiation mitigation measures are required to ensure secure functionalities of on-board C&DH components.
  - Implementing redundancy contributes to improving the system reliability.

Example of a tightened harness system against launch vibrations

Example of a copper heat-path from an IC chip to a mechanical fastener.



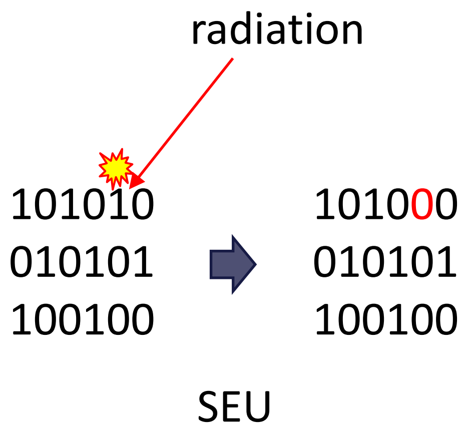
Example of a redundant computer unit, consisting of nominal and redundant computers.

4. Space Environmental Effects on C&DH Subsystem

4.2. Radiation Effects on C&DH Electrical Components

Classification of Radiation Effects:

- Soft Error: Can be recovered by power cycling. *Temporal Effect.*
- Hard Error: Hardware degradation and/or damage. *Permanent Effects!*



Type of Effects:

- Total Ionizing Dose (TID) effects [Hard Error]
  - total sum of radiation hitting the target component during mission time.
- Single Event Effects (SEE)
  - triggered by deposited energy from charged particles traversing through electrical devices.
  - Single Event Upset (SEU) [Soft Error]:

    The energy transferred by charged particles results in state changes of flip-flops (1 changes to 0 and vice versa).
  - Single Event Transients (SET) [Soft Error]:

    Generated temporal current flow on conductive parts in electrical devices, which can interact with clock signals by widening the pulse signal by falling on the trailing edge and narrowing it down by falling on the front edge of the signal.
  - Single Event Latch-up (SEL) [Hard Error]:

    Excessive current flow induced by sufficient energy from a charged particle and consequently to a permanent loss of device functionality

4.3. Computer System Design Strategy against Radiation Effects

- SRAM or RAM devices are supposedly easily affected by radiation, especially soft errors such as SEU.
  - This means that systems controlled by CPUs in a software manner have the potential of being affected by radiation, because of their use of program memories, data memories, cache memories etc.
  - All these volatile memories are vulnerable against radiation.
  - A single bit of SEU error can lead to critical loss of functionality.
- A hardware system without memory elements is very immune against radiation. The key technique in establishing radiation-tolerant or radiation-hard electronic systems is to conduct clear categorization of functionalities into;

  1) the ones which can be temporarily affected by radiation effects in a soft-error manner, and

  2) the ones which cannot fail during the whole mission lifetime.
- As much as possible, 1) can be implemented into software-based system, and 2) should be implemented into hardware system.
- This trade-off between hardware and software plays significant role in realizing a higher system reliability against radiation effects.
- It is very important that the satellite system is implemented with a power cycling capability for the temporally failed components.

4. Space Environmental Effects on C&DH Subsystem

4.4. Radiation Tolerance Characteristics of FPGA devices

- FPGA: Field Programmable Gate Array
  - Radiation tolerance depends on type of internal structure

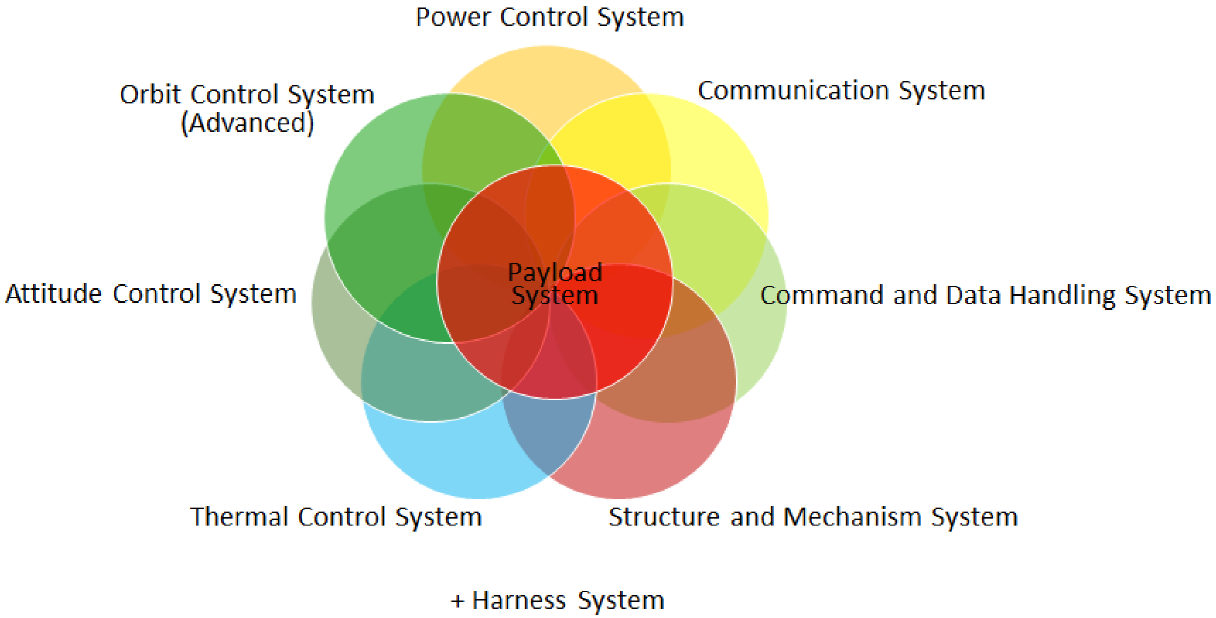
- SRAM-FPGA
  - Internal SRAM elements tend to be vulnerable against single event effects. Requires power cycling to repair temporal radiation effects. Devices with a high TID tolerance are available.
- Flash-FPGA
  - Internal logic elements are radiation tolerant (non-volatile). Temporal memory elements are vulnerable against single event effects and shall be protected against radiation, e.g., via multi module redundant design.
- Antifuse-FPGA
  - Internal logic elements are radiation hard. Temporal memory elements of space-grade devices are radiation protected by design through Triple Module Redundancy (TMR).

It is important to select appropriate FPGA devices according to the mission objectives and required reliability!

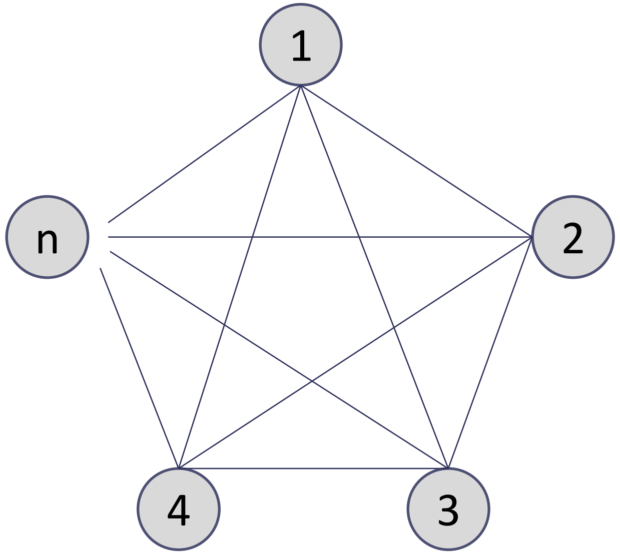
5. CubeSat System Integration

5.1. Interface between Subsystems

- A system with n components has $\frac{n(n-1)}{2}$ interfaces in between.
- For CubeSat system integration, the interfaces between subsystems, as well as components, shall be clearly defined.



Satellite Subsystems



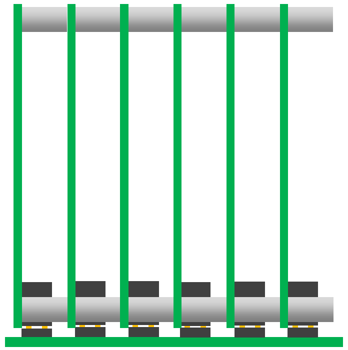
$$Number\ of\ Interface = \frac{n(n-1)}{2}$$

System Interface

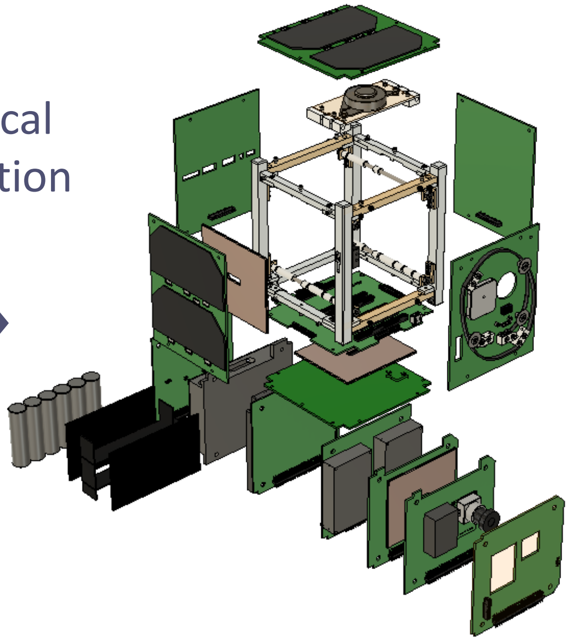
5. CubeSat System Integration

5.2. CubeSat Integration Process Overview

- Thanks to worldwide CubeSat interface standardization efforts, the mechanical and electrical systems of CubeSats can be integrated in a very dense manner to a limited mechanical envelope.
- System design and integration shall be planned in the way that the resources for the payload can be maximized, especially such as the mass, envelope, and power.
- CubeSats need to be assembled to fulfil the tolerance requirements of mechanical dimensions to fit in the pod.

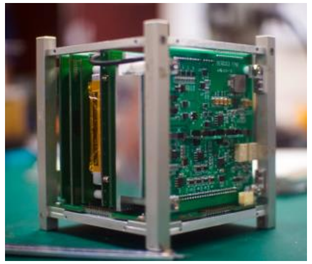


Electrical Integration



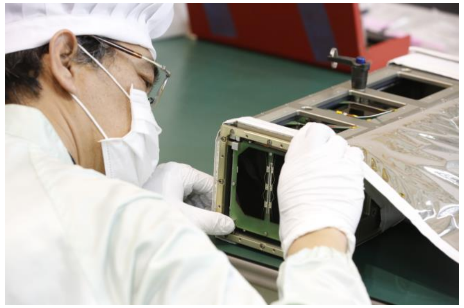
© Kyutech

Mechanical Integration



© Kyutech

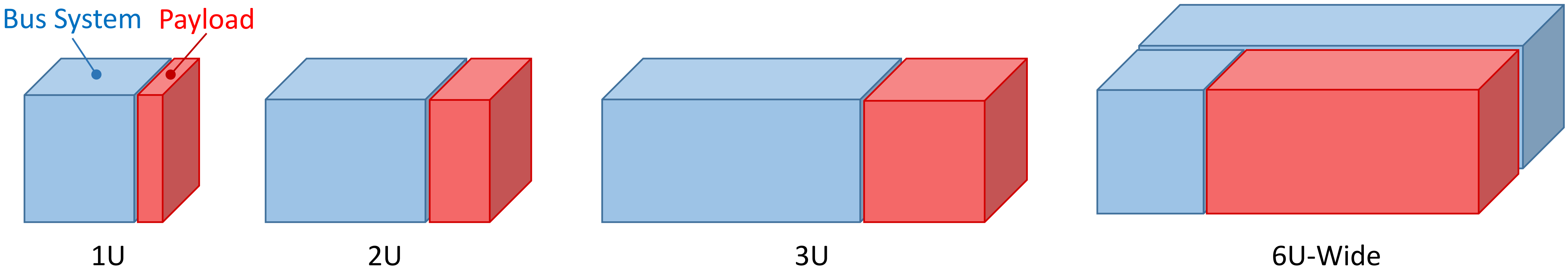
Assembly into Release Pod



© JAXA

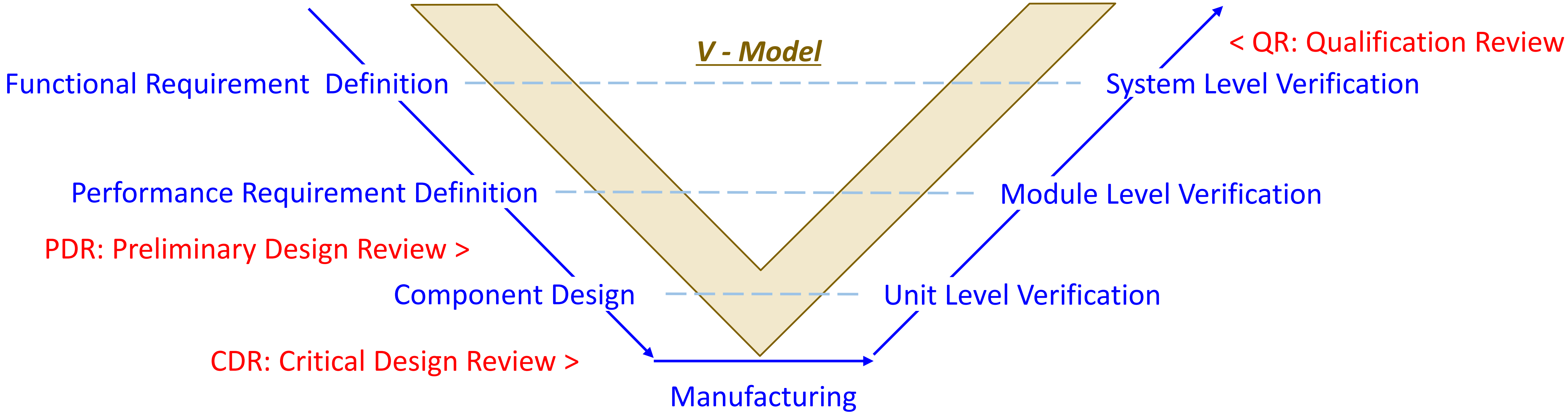
5.3. Relationship of Payload and Bus System

- The size of the CubeSat for each mission shall be carefully selected based on the difficulty level of the mission objective and the complexity of the requirements on payload instruments.
- Larger, high-performance bus systems will be required for high-level missions.
  - Higher power generation, storage, and consumption capability => Large solar panels and more battery capacity
  - Higher data processing capability, more memory storage, higher communication throughput.
  - Accurate attitude determination and control capability.
- The larger and the more complex the system is, the more effort will be required for the system integration.
- It is recommended that the interface between the payload and bus system is clearly defined so that they can be developed in parallel to shorten the development and verification time schedule.



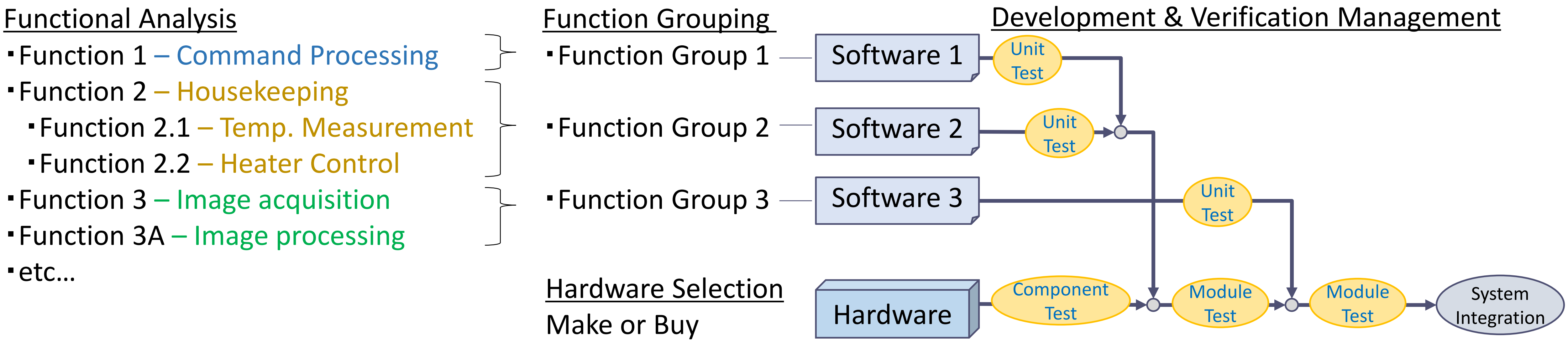
5.4. Development and Verification of Satellite Components

- The requirements of the satellite components shall be defined and driven from the mission objectives and mission requirements. Component requirements include functional requirements, performance requirements, constraints, etc., both for the related hardware and software.
- Requirements shall be detailed by a preliminary design review, and component design shall be defined by a critical design review. Components shall be verified in a “bottom-up” manner, followed by a qualification review.



5.5. Hardware and Software Development and Verification Process

- Software functionalities shall be analyzed as the first step and functional requirements shall be identified.
- Functions can be classified into some functional groups which are implemented as a software unit for the component.
- Development schedules and progress of each hardware and software element shall be managed, and the order of their integration and verification shall be defined in order to control the development schedule.
- Unit tests can be at any level and any kind, in order to ensure that the components fulfill the requirements.
- Identification of verification items is critically important for the mission success. *Need experience!*

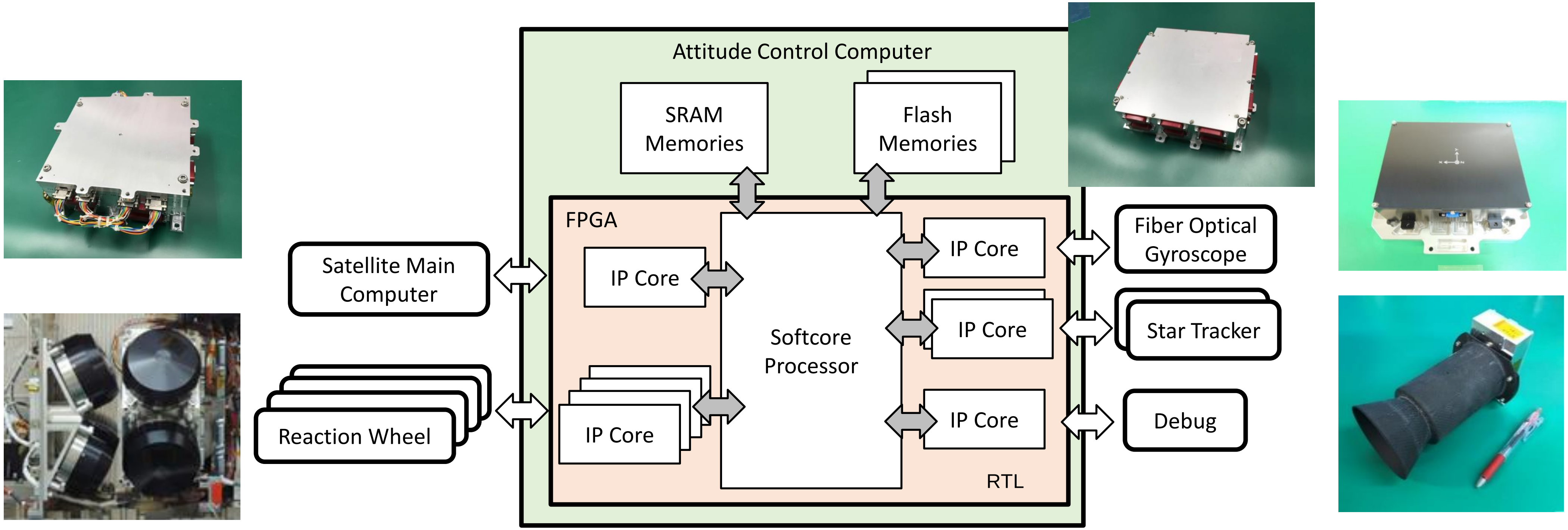


5. CubeSat System Integration

5.6. System Integration through System-on-a-Chip Implementation

- A computer system can be constructed through a System-on-a-Chip (SoC) design approach. Recent advanced of FPGA (Field-Programmable Gate Array) technology allows you to implement CPU and custom peripheral interface logics inside a single FPGA chip.
- A large number of peripheral components can be connected to the computer.

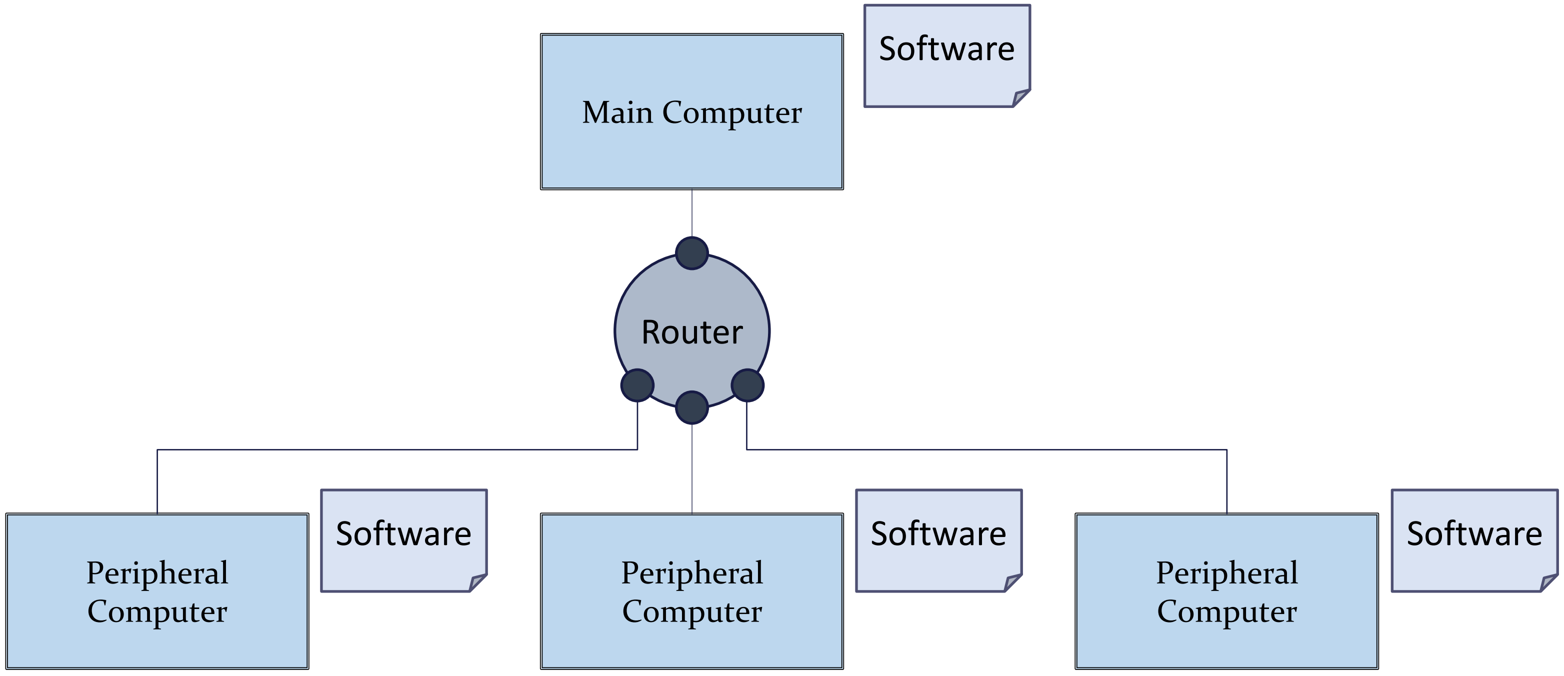
System-on-a-Chip Design Approach – An example of Micro-satellite (RISESAT) Attitude Control System



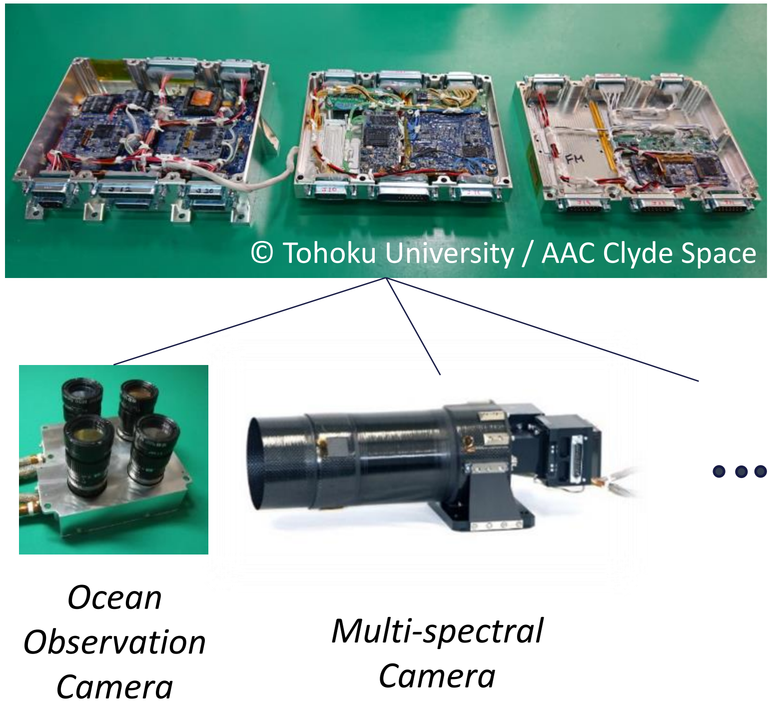
5.7. System Integration through Plug-and-Play

- Standardization of hardware and software can realize satellite system integration through Plug-and-Play (PnP).
- There are several PnP standards suggested for several types of interfaces, such as I^{2}C, USB, SpaceWire, etc.
- Through PnP technology, one can minimize the system integration effort and maximize reusability.

Plug-and-Play Architecture



PnP Computer of Micro-satellite RISESAT



© Hokkaido University / Tohoku University

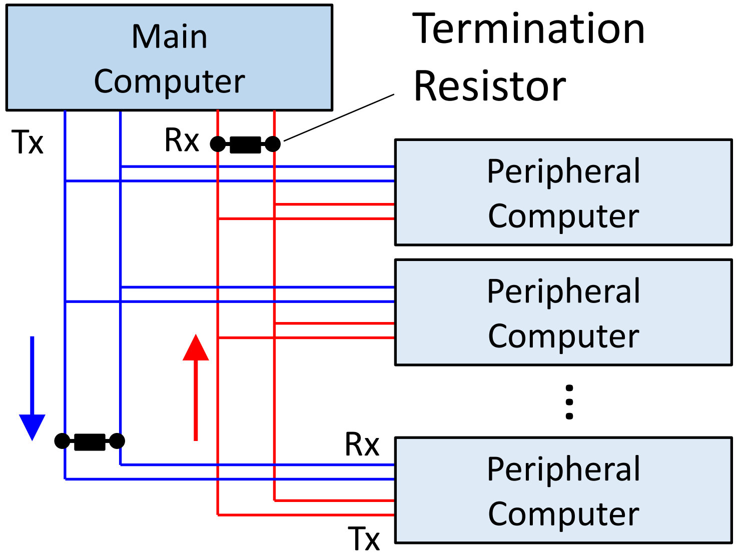
5.7. System Integration through Plug-and-Play

- In order to apply Plug-and-Play (PnP) types system integration methodology, software functionalities are required, such as self-identification, configuration, common language standard, device driver, etc.
- Example： Space Plug and Play Avionics (SPA) Standard by the AIAA
- Characteristics of PnP Implementation:
  - More development effort may be required in order to provide versatility for the first-time-implementation.
  - Beneficial if more than one satellite will be developed, or satellites are mass-produced.
  - Electrical interface (connectors, pin assignments, etc.) needs to be standardized.
  - Power supply specifications need to be defined in a common form, which might lead to a difficulty in setting electrical device-specific custom parameters, such as the current threshold, to optimal values.
  - Degree of freedom in hardware assembly can be increased with standardized hardware interfaces.

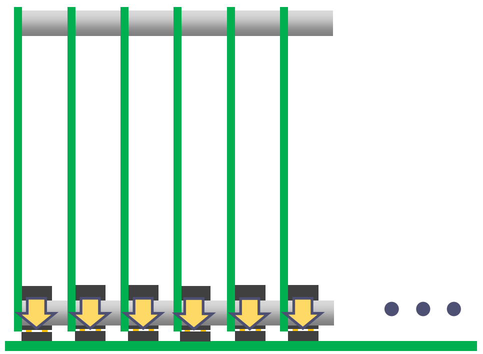
5.8. System Integration through Bus Communication Interfaces

When using bus communication standards with electrical multi-drop connections, the following aspects shall be considered:

- Termination resistors should be installed on the device furthest from the master device.
- In the case where electrical boards are identical, the termination resistors can also be mounted on the backplane at the root of the last board.
- The communication throughput may become low because the communication is sent and received over shared communication lines.
- Fast enough for command and telemetry data exchange.
- Dedicated communication lines might be necessary for exchanging large amounts of data between boards, such as payload image data.



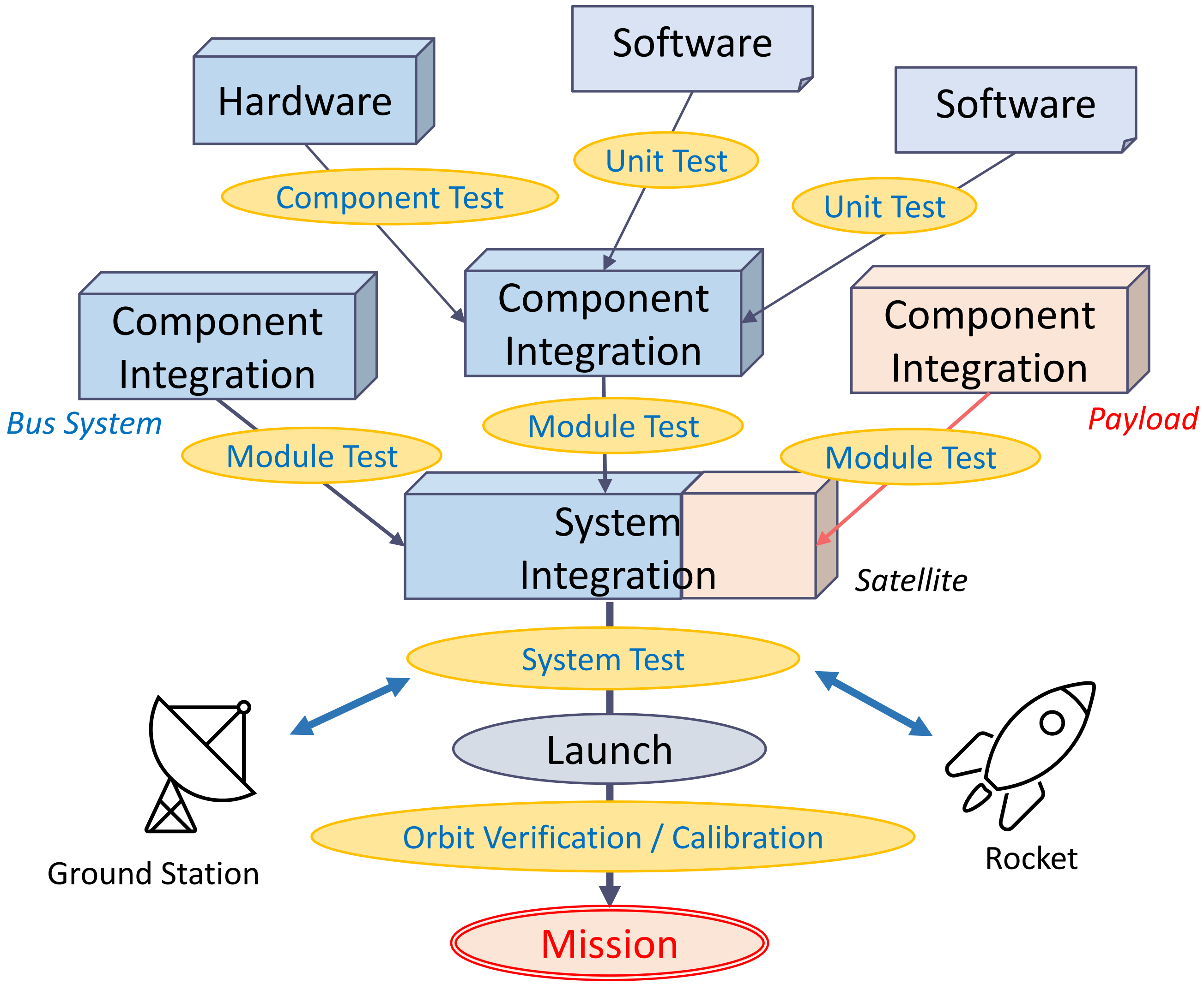
Bus Architecture



Backplane Style

5.9. Satellite System Integration and Testing

- Satellite system integration takes place in a bottom-up manner, starting from the hardware and software integration of each component, integration between components, and integration between the bus system and payload.
- Assembly and testing shall be conducted in each integration test. This activity is sometimes referred as Assembly, Integration, and Test (AIT).
- The scope of the system level testing shall include testing together with the ground stations and launch vehicles, or its interfaces.
- Satellite hardware and software functionalities shall be tested and calibrated even after the launch, in order to ensure that the satellite can fulfil the mission requirements.



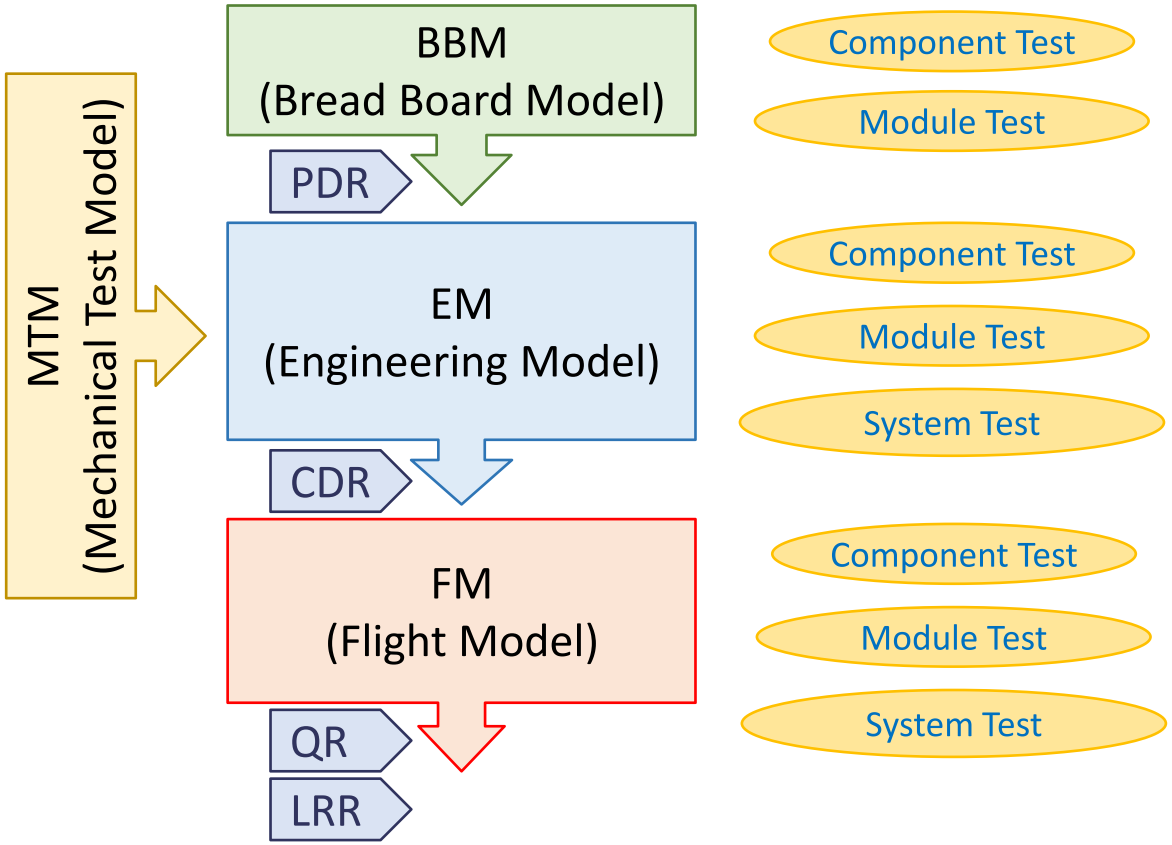
6. Functional Verification of C&DH System

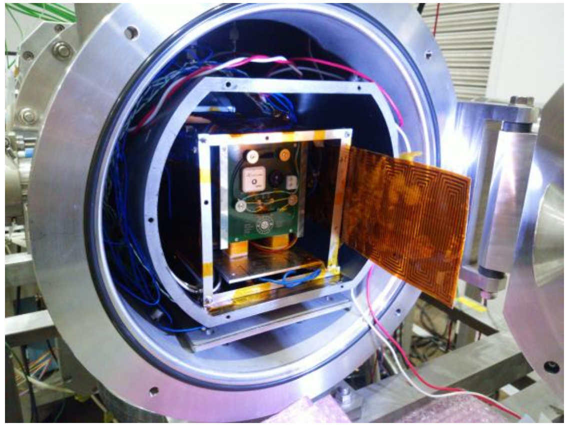
6. Functional Verification of C&DH System

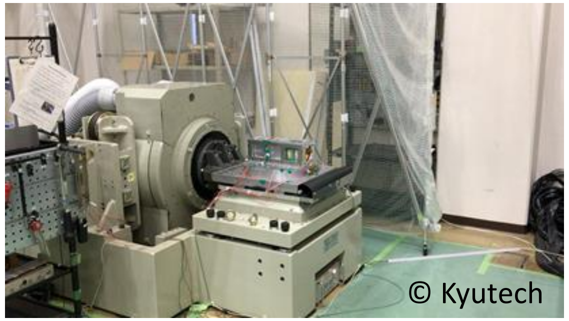
6.1. Verification Process

- Verification processes of a satellite can start from the Bread Board Model (BBM) concept verification.
- Detailed design solution of the satellite is verified using Engineering Model (EM).
- Flight Model (FM) is manufactured based on the verified design through EM, and minimum required tests are applied to obtain qualification for the launch.
- Sometimes a mechanical test model is utilized before the manufacturing of the structure of the EM.

Satellite Verification Process



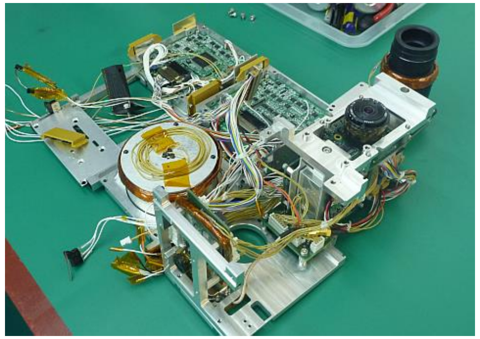


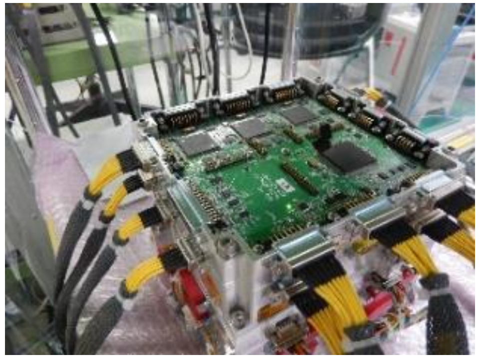


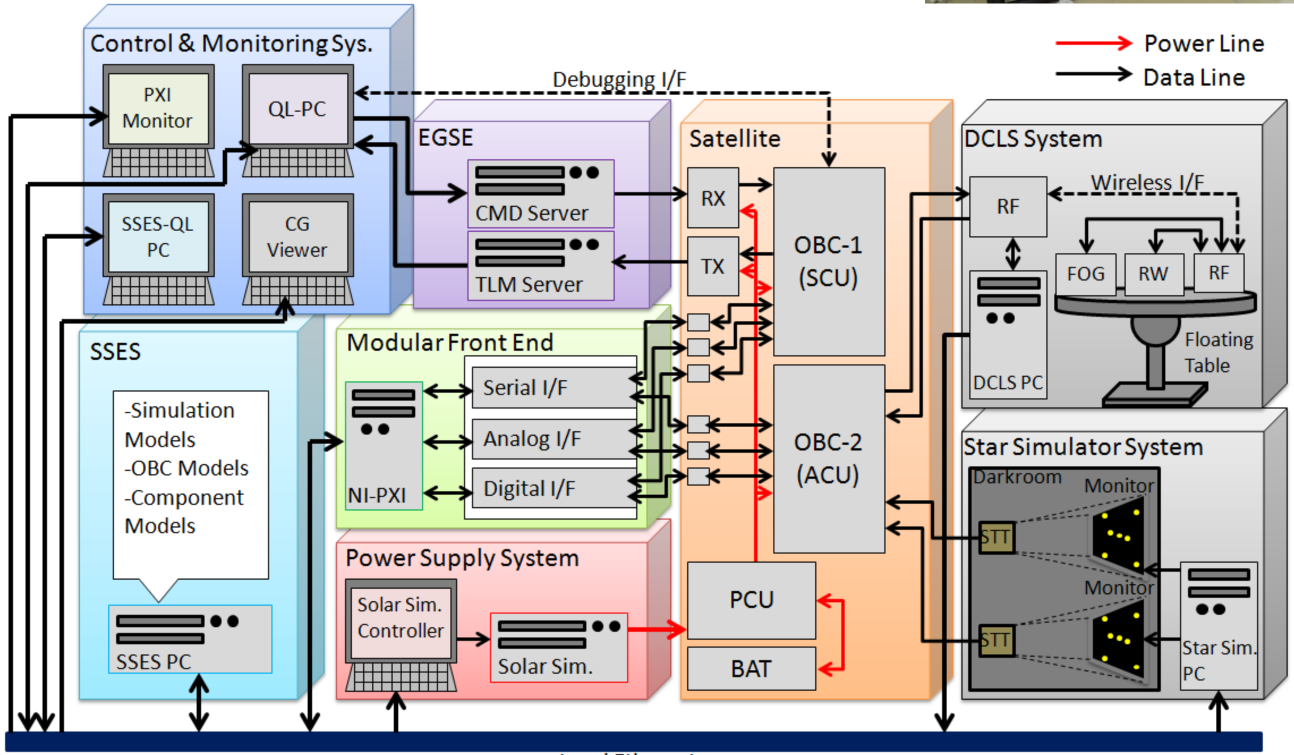
6. Functional Verification of C&DH System

6.2. Electrical Testing

- Ground testing facilities are required to conduct tests of electrical functionalities of the satellite components and satellite system.
- Software-based simulators are utilized for hardware-in-the-loop tests in real-time in order to verify the correct functionalities of on-board software running on the actual flight hardware.

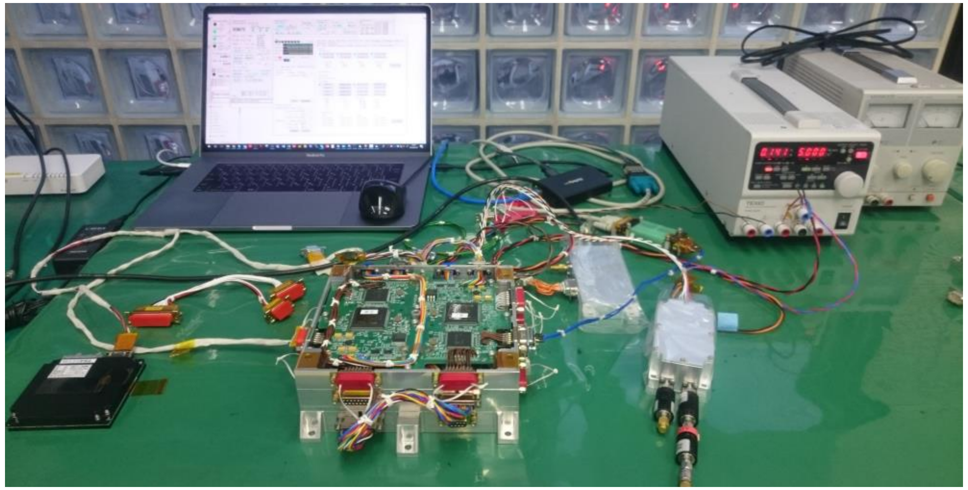






6.2. Electrical Testing – “Flat-Sat” configuration

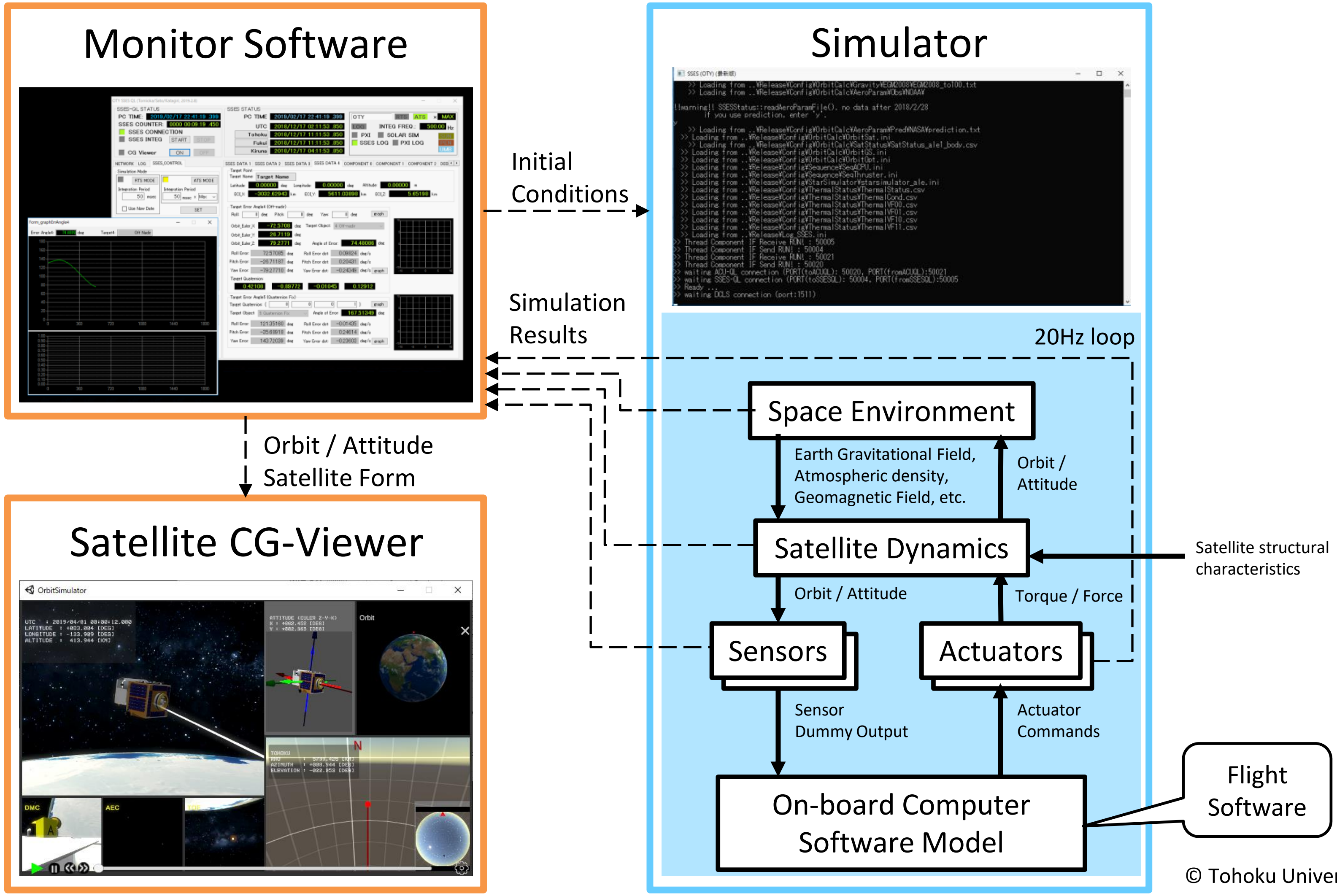
- Unit test
  - Unit tests can be conducted at first by connecting the device with the simulation environment or emulators of peripheral devices.
- Interface Test
  - Once each single unit has been verified, multiple devices are connected, and electrical compatibility and communication compatibility are tested.
  - To check the communication interface with the ground station, the on-board communication device and OBC are connected, and RF cable connections are used to connect to the ground station to conduct an RF tests as illustrated in the above figure.
- System Test
  - On-board electrical components are connected and tested one by one.
  - Since the power supply system and harnesses are often connected to the satellite structure, testing is often performed while the electrical components are actually being integrated to the satellite.
  - In addition to end-to-end tests, long-term operation demonstration tests in this configuration, controlled from the ground station equipment, contribute to increase the system reliability, as illustrated in the bottom figure.



6. Functional Verification of C&DH System

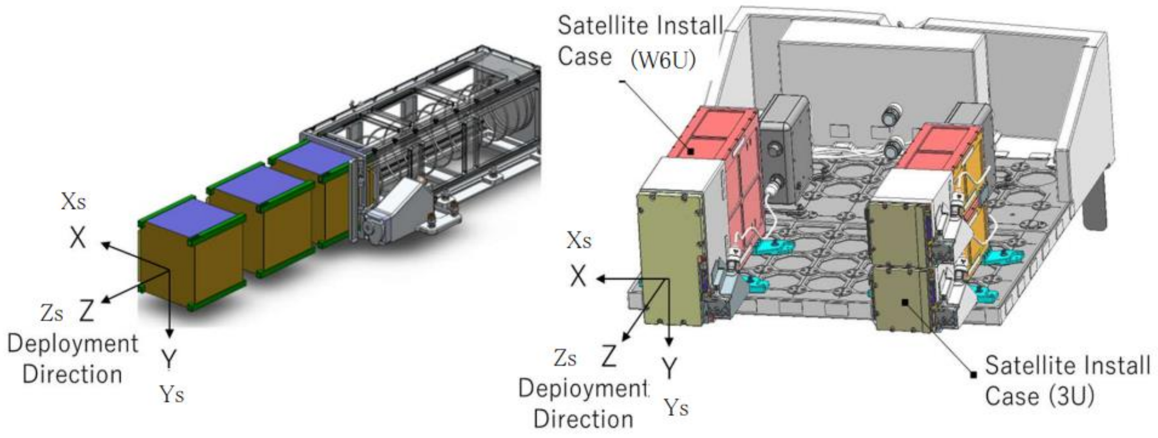
6.3. Software Simulation

- Full software simulators are very useful for the simulation of the satellite's orbital and attitude behavior.
- The simulation process can be accelerated to conduct a large number of simulation trials.
- On-board software can be developed using this kind of simulation, software development, and verification environment.



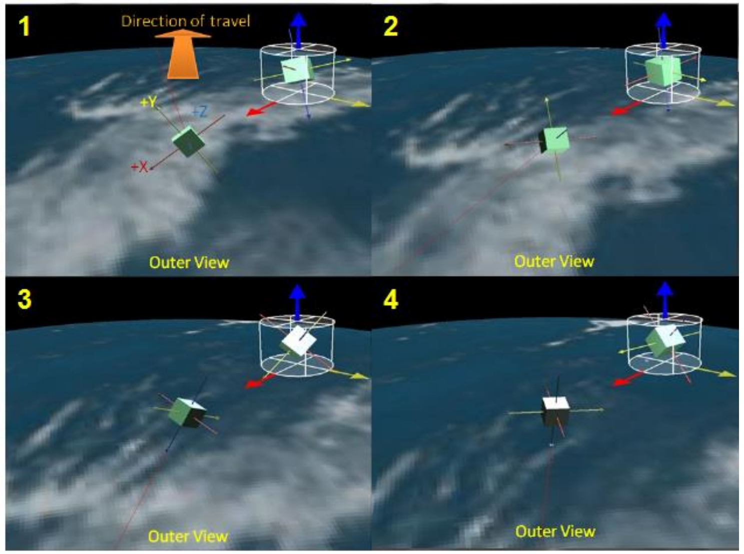
6.4. Fit Check

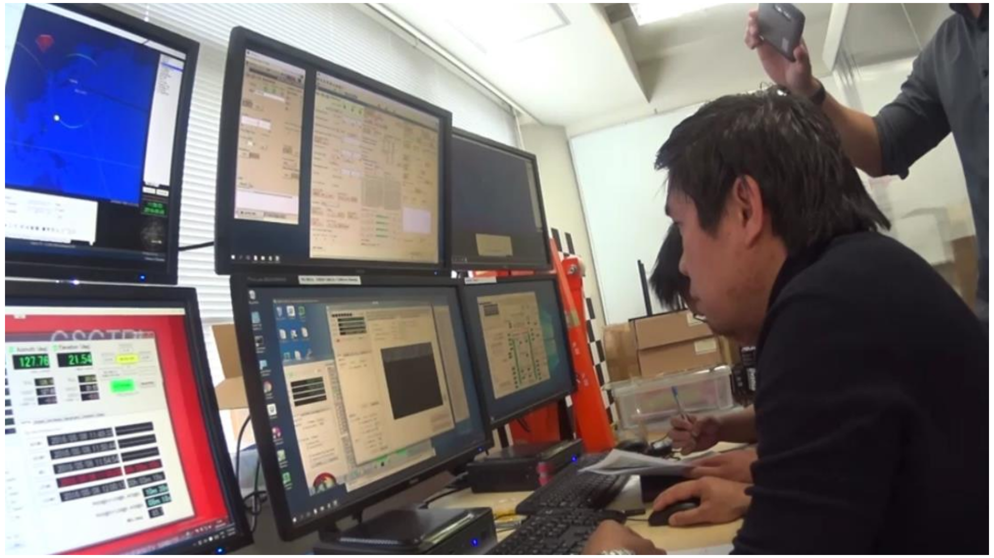
- "Fit check" is a one of the most important tests of a CubeSat. A CubeSat's mechanical and electrical compatibilities are tested using a mechanical Fit Check model of the CubeSat deployment pod.
- The manufacturing and integration of the mechanical system shall fulfill the mechanical requirements posed on CubeSats, such as dimension tolerance, surface accuracy, contact of mechanical switches, etc.
- Not only the structural design, but also the satellite integration process, shall be planned in the way that the satellite assembly and integration process is reproducible.
- It is very important that the assembly, integration and test results are carefully recorded in documents.



6.5. Operation Training

- Operational training shall be started at an early stage of the satellite development, so that the verified software and procedure can be reflected to the flight software.
- Ground station software and on-board software shall be developed and tested in parallel.
- Satellite operation planning skills shall be established within the team using the environment.
- Mission lifetime of a satellite is limited, and hence, effective operation of the satellite is indispensable in order to ensure a secure operation of the satellite and to obtain maximum achievements.





© Tohoku University

7. Conclusion

7. Conclusion

- CubeSat Command & Data Handling (C&DH) system were introduced. Three different types of data processing were described: command processing, on-board data handling, and autonomous functions.
- Hardware components of CubeSat C&DH system were introduced. CubeSat-related standards were introduced, launch opportunities and available mechanical form factors were described. The CubeSat system interface was introduced in terms of electrical interconnections between satellite components, system block diagrams, system configuration design, power distributions, communications, and mechanical interfaces.
- Software development and integration aspects of CubeSat C&DH system were described. Operational planning and command list preparation procedures were also discussed.
- System integration and verification processes, as well as their engineering aspects, were described both in hardware and software aspects. Some related advanced topics were introduced, such as the System-on-a-chip design method, system integration based on Plug-and-Play, and bus communication interfaces.
- Space environmental effects on C&DH system were explained. Radiation mitigation methods were introduced.
- CubeSat functional verification methods were introduced. Topics such as environmental testing, electrical functional verification, fit check with the deployment pod, and operation training were discussed.

Thank you very much.

[Disclaimer]
The views and opinions expressed in this presentation are those of the authors and do not necessarily reflect those of the United Nations.