Open-source Software Suite for Small Satellites: C2A (Command Centric Architecture), S2E (Spacecraft Simulation Environment), and WINGS (Web-based Interface Ground-station Software)

Ryo Suzumoto, Satoshi Ikari, Kota Kakihara, Takayuki Hosonuma, Kanta Yanagida, Yuta Sakamoto, Masahiro Fujiwara, Hirotaka Kondo,
Kentaro Enokida, Toshihiro Suzuki, Yoshinari Gyu, Hirotaka Sekine, Tomoki Mochizuki, Yutaro Ito, Ryu Funase, Shinichi Nakasuka

ArkEdgeSpace
THE UNIVERSITY OF TOKYO

✉ suzumoto@space.t.u-tokyo.ac.jp / suzumoto@arkedgespace.com

Abstract

Small satellites are becoming more and more widespread, not only for technological demonstration missions but also for actual practical missions. Many small satellite projects have tight budgets and schedules, while the complexity of the missions is increasing. In such a situation, the technical development of satellite software is very important. We made the following software, essential for satellite development, open source at the ISSL (the Intelligent Space Systems Laboratory at the University of Tokyo) Organization on GitHub[1]: a flight software C2A (Command Centric Architecture), a satellite simulator S2E (Spacecraft Simulation Environment), and a ground station software WINGS (Web-based Interface Ground-station Software). We are also trying to utilize software engineering knowledge outside the satellite industry and develop them further.

Flight Software: C2A

ISSL has been developing an onboard software architecture called C2A (Command-Centric Architecture) since around 2013; its development has been based on ISSL's experience with satellite development and operation. C2A has been developed with a focus on providing high reusability and flexible on-orbit reconfiguration capability.

C2A has two parts: "C2A Core", which consists of the flight software kernel code and mission- and hardware-independent code, and "C2A User", which is developed for each mission and OBC[2]. The flight software developer of each OBC develops their software for each mission based on the "C2A User Sample". As shown by the red vertical arrows in Fig. 2, when C2A Core is updated on the mainline, the developer can use the new C2A Core by updating it of their own C2A. In this way, the source code of the kernel part of C2A can be shared directly with various flight software as C2A Core. In addition, C2A has high portability and can run on various MPUs such as STM, PIC, and SH, as well as OS such as Windows and Linux. Because of these characteristics, C2A has been used not only in the Main OBC, the AOCS OBC, and the Thermal OBC of the "ONGLAISAT" satellite developed by ISSL, but also in OBCs used in other organizations, such as the "MAGNARO" satellite developed by Nagoya University and the "AE1a" satellite developed by ArkEdge Space Inc.

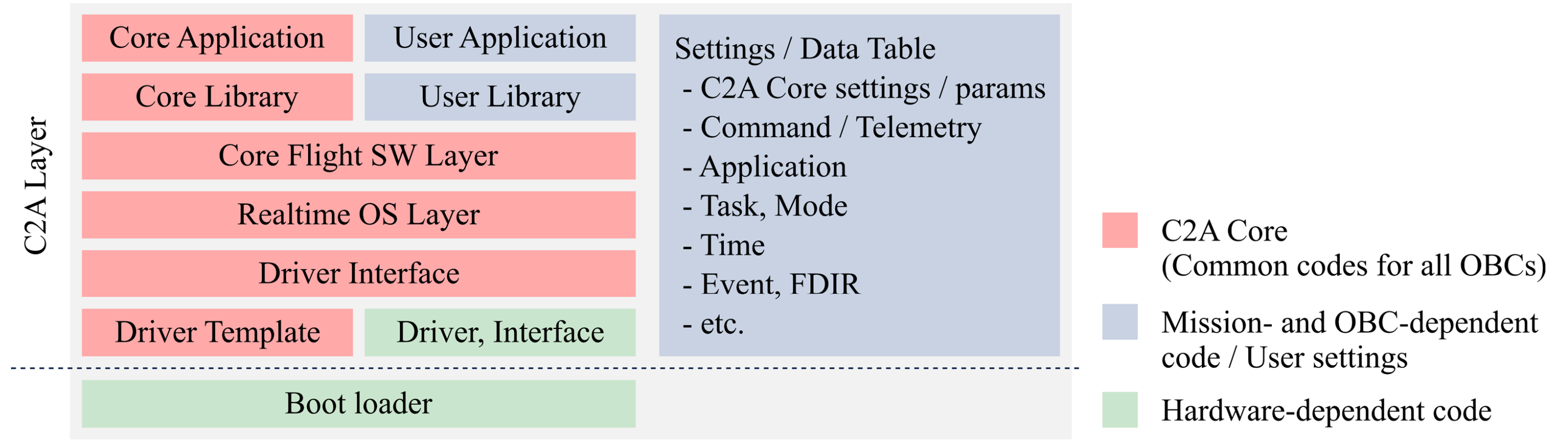


Fig. 1. Schematic Diagram of C2A Core and C2A User Code

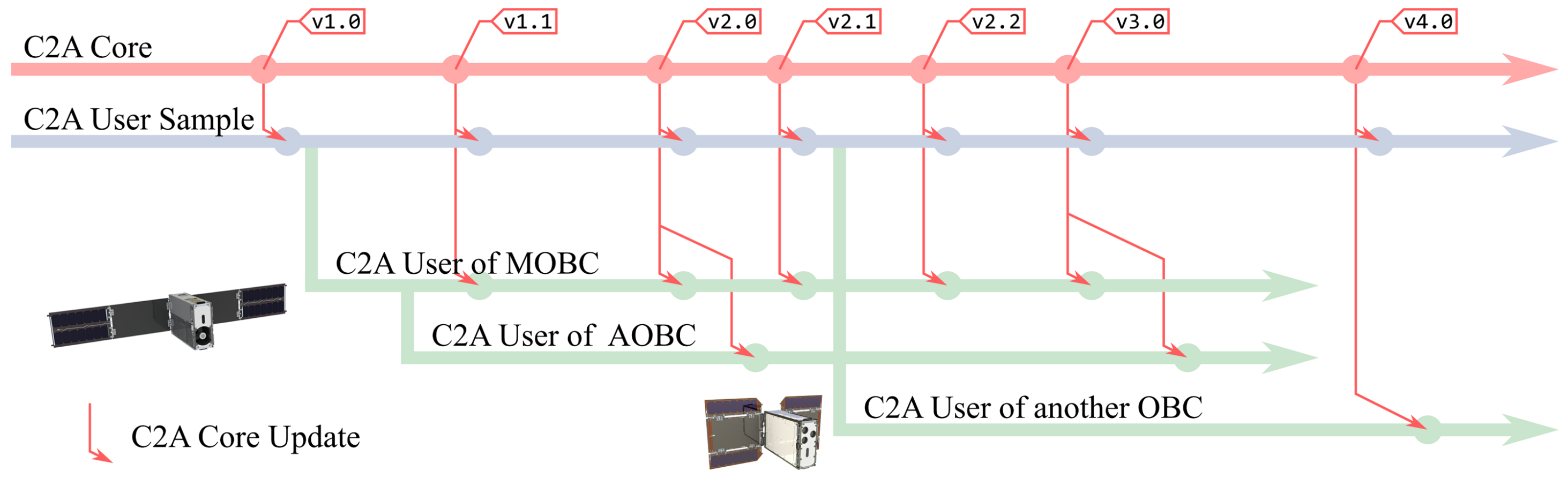


Fig. 2. Flight Software Development Using C2A

Satellite Simulator: S2E

S2E (Spacecraft Simulation Environment) is a satellite numerical simulator developed based on the "Nano-JASMINE" satellite's simulator, combined with the launch experience of the "PROCYON" spacecraft. The basic feature of S2E is to simulate satellites' orbit- and attitude-behavior. In addition, to improve the fidelity of the simulation, S2E can calculate space environments (e.g., celestial body position, geomagnetic field) and orbital disturbances (e.g., air drag, high-order geopotential). All simulation features required for typical satellite analysis are implemented so that S2E can be used for satellite attitude and orbit control research. Furthermore, S2E can simulate communication with onboard satellite components such as sensors and actuators and power supply for these components. By combining S2E with flight software such as C2A, more realistic satellite software verification tests and operational analysis can be performed.

The features commonly required by many users are released as "S2E Core", and users implement their own simulation scenario called "S2E User" by referring to the Core code. ISSL utilizes S2E as a simulator for formation flying control research, high-precision orbit determination research, and attitude control verification tests to develop the integrated attitude control units.

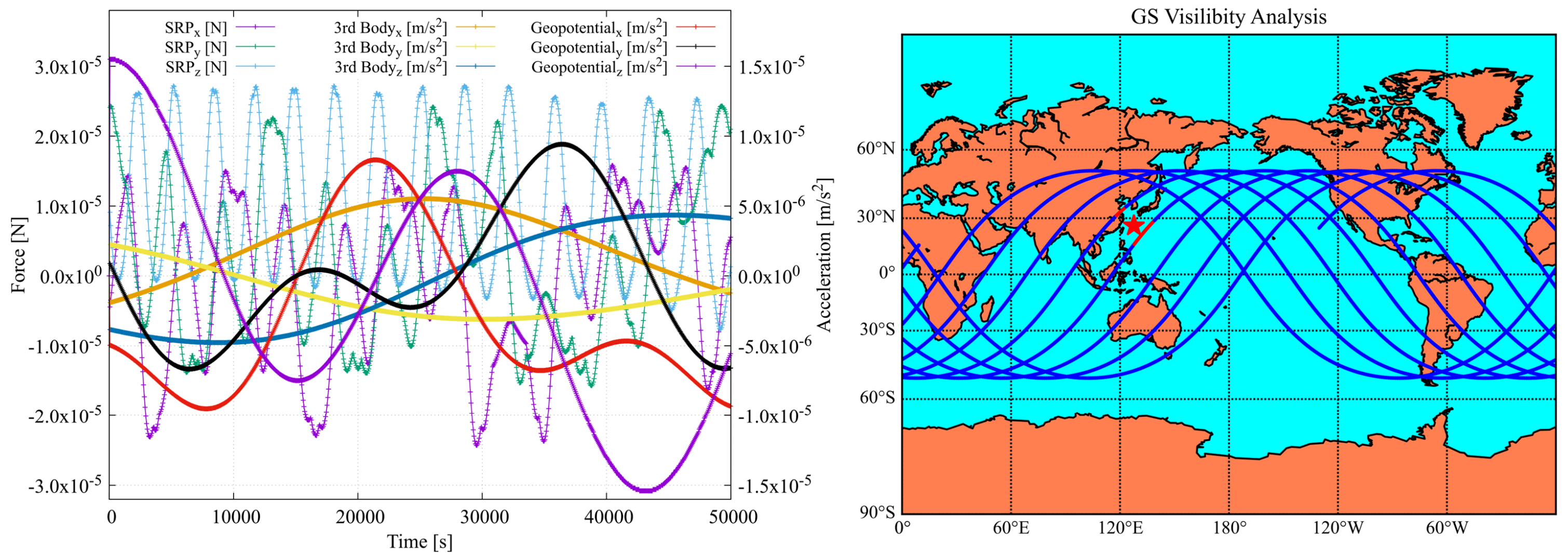
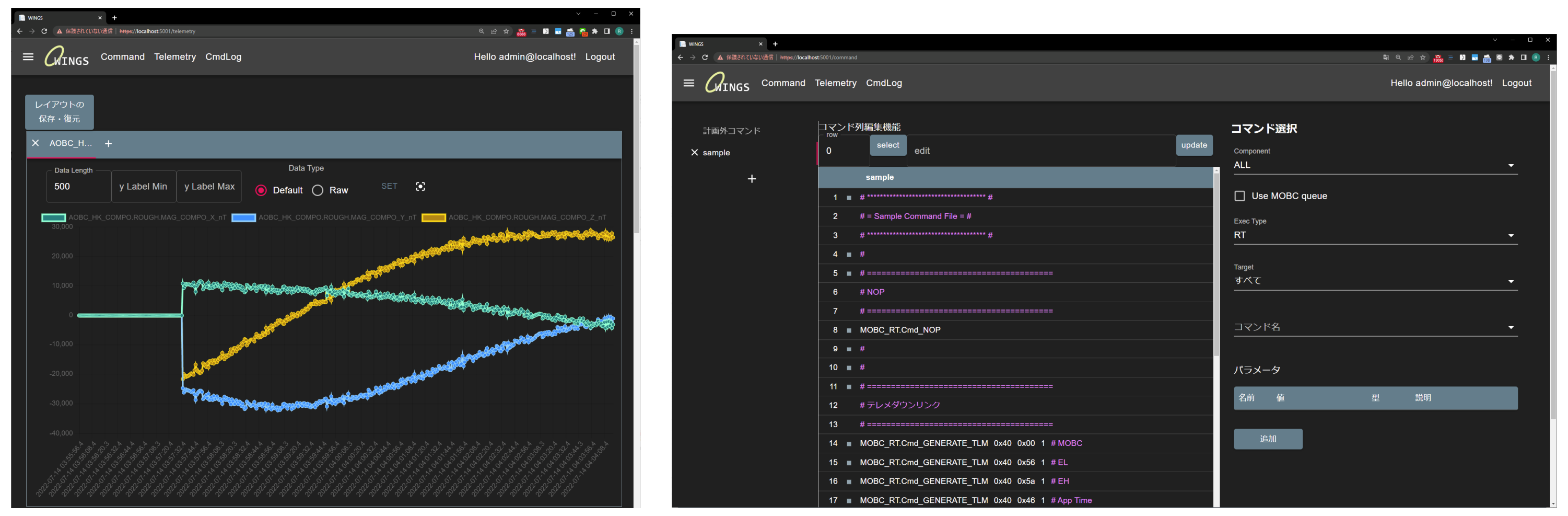


Fig. 3. Simulation Results by S2E

Ground Station Software: WINGS

WINGS (Web-based Interface Ground-station Software) is a web-based ground station software developed since 2020. WINGS has been developed for use in satellite operations, ground tests, and component tests. The front-end and back-end of WINGS are implemented separately, and the front-end calls REST APIs to connect the back-end. Therefore, the user interface can be extended to be more user-friendly by modifying only the front-end. Furthermore, by calling the APIs via scripts such as Python, it is easy to perform automated testing of components and build an automated operating system. The target communication devices of WINGS are not limited to onboard satellite transmitters and receivers; WINGS can communicate with any onboard satellite component by writing driver code, whose operations can be performed using a Web browser. WINGS is developed in ASP .NET and can run on multiple platforms including Windows, Mac, and Linux.

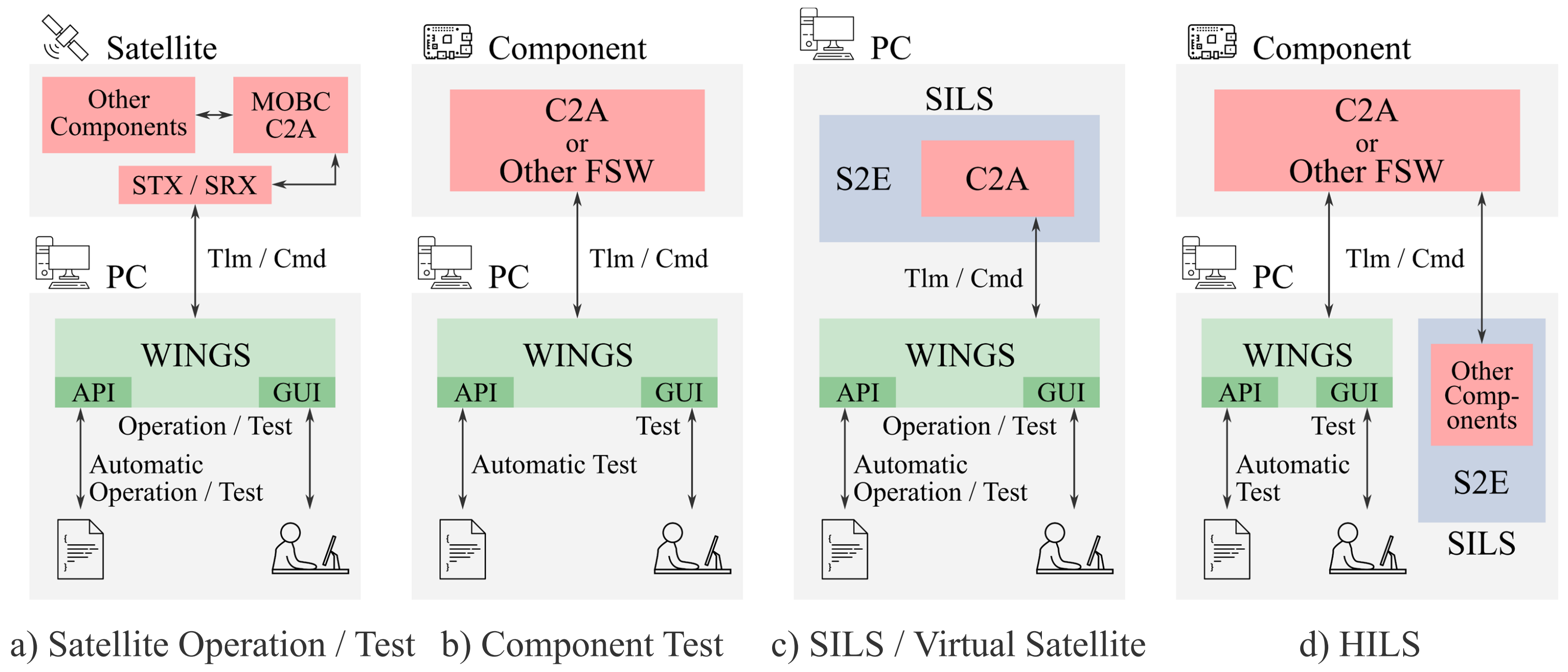


```
sample
1  # ********************************* #
2  # = Sample Command File = #
3  # ********************************* #
4  #
5  # ====================================
6  # NOP
7  # ====================================
8  MOBC_RT.Cmd_NOP
9  #
10 #
11 # ====================================
12 # テレメダウンリンク
13 # ====================================
14 MOBC_RT.Cmd_GENERATE_TLM  0x40  0x00  1  # MOBC
15 MOBC_RT.Cmd_GENERATE_TLM  0x40  0x56  1  # EL
16 MOBC_RT.Cmd_GENERATE_TLM  0x40  0x5a  1  # EH
17 MOBC_RT.Cmd_GENERATE_TLM  0x40  0x46  1  # App Time
```

Fig. 4. WINGS GUI

Collaboration of Each Software

By combining C2A, S2E, and WINGS, various things can be achieved to increase the efficiency of satellite development and research. WINGS is available to connect to a satellite for operating and testing (Fig. 5 a). WINGS is also available for stand-alone tests of components (Fig. 5 b). In this way, WINGS can be used as a unified interface to handle satellites and their components. By executing C2A on S2E, actual flight software and satellite environment can be simulated simultaneously as SILS (Software In the Loop Simulator), and the virtual satellite can be operated via WINGS (Fig. 5 c). This virtual satellite operation environment allows users to experience satellite operations and can be used for education and research. In our flight software development, when new source code is implemented, this environment is automatically launched on the CI (Continuous Integration) server, and various tests are running to automatically verify the flight software[2]. In addition, by integrating an actual OBC with S2E, HILS (Hardware In the Loop Simulator) can be constructed to perform more realistic verification (Fig. 5 d).



a) Satellite Operation / Test b) Component Test c) SILS / Virtual Satellite d) HILS

Fig. 5. Various Configurations Using This Software Suite

OSS Activities

We have open-sourced C2A, S2E, WINGS, and their related software and tools under the MIT license. Currently, ISSL, with its experience in developing advanced satellites and related research, and ArkEdge Space Inc., with its extensive expertise in satellite mass production and software development, continue to work together to develop this software suite. By making software open source, universities and private companies can cooperate with each other by sharing their strengths, and high-quality software is being developed quickly. In recent years, the number of inquiries about this software suite has been increasing, and the number of users has been steadily growing. Furthermore, this software suite has come to play a successful role in connecting research and satellite development.

References

[1] GitHub Organization of Intelligent Space Systems Laboratory, The University of Tokyo, https://github.com/ut-issl

[2] Ryo Suzumoto, *et al.*, Improvement of C2A (Command-Centric Architecture) Reusability for Multiple Types of OBCs and Development of Continuous Integration Environment for Reliability of Flight Software, 33rd ISTS, 2022.

[3] Satoshi Ikari, *et al.*, Development of Compact and Highly Capable Integrated AOCS Module for CubeSats, 33rd ISTS, 2022.