CubeSat bus interface with Complex Programmable Logic Device

CubeSat bus interface with Complex Programmable Logic Device

Turtogtokh Tumenjargal[*], Sangkyun Kim, Hirokazu Masui and Mengu Cho

Laboratory of Spacecraft Environment Interaction Engineering,
Kyushu Institute of Technology,
1-1 Sensui, Tobata-ku, Kitakyushu 804-8550, JAPAN

Abstract

A standardized interface for different CubeSat missions is one of the keys to reducing costs and delivery time. A backplane interface approach, proposed by the University of Würzburg in Germany as UWE-3, was implemented in three CubeSat projects at the Kyushu Institute of Technology (Kyutech) to shorten the development and assembly times. The backplane approach also helped to reduce the risk of workmanship errors associated with the harness. However, changes to the proposed standard interface board were necessary in every CubeSat project, to comply with the mission requirements. To obtain more flexibility, especially for data connections, this work introduces a novel idea of a software-configurable bus interface with a backplane board. A Complex Programmable Logic Device (CPLD) was used instead of hardware routing so that we can reconfigure the bus interface by reprogramming the CPLD. The concept was validated by a functional test with a breadboard module. A radiation test verified that the selected CPLD has enough strength to survive total ionization doses of more than 2 years in low Earth orbit. A new backplane board with CPLD has been integrated into the engineering model of the fourth CubeSat project at Kyutech, the BIRDS-3 project, and system level verification has been conducted.

Key words: CubeSat, Interface standard, Bus interface, BIRDS Project, CPLD

Acronyms/Abbreviations

The following acronyms and abbreviations are used in this paper:

ANT – Antenna System Board	OBC – On Board Computer
BBM – BreadBoard Module	PC – Personal Computer
CAN – A Controller Area Network	PCB – Printed Circuit Board
COM – Communication Subsystem	RAB – Rear Access Board
CPLD – Complex Programmable Logic Device	SEL – Single-Event Latch-Up
EM – Engineering Model	SEU – Single-Event Upset
EPS – Electric Power System	SoftCIB – Software Configurable Interface Board
FAB – Front Access Board	SPI – Serial Peripheral Interface
FPGA – Field Programmable Gate Array	TID – Total Ionizing Dose
FT – Functional Test	TVT – Thermal Vacuum Test
GS – Ground Station	UART – Universal Asynchronous Receiver-Transmitter
I2C – Inter-Integrated Circuit Bus	UHF – Ultra High Frequency
ISO – International Organization for Standardization	
MSN – Mission Board	

[*]*Corresponding author. Tel.: +81-093-884-3229; Email: p595906t@mail.kyutech.jp*

1. Introduction

The number of small satellite launches per year and the number of players in this field have been rapidly growing since the 2000s, which is strongly related to the popularity of the CubeSat standard [1] around the world [2,3]. The main reasons for its successes are often related to its launch compatibility through a standardized deployer mechanism, (i.e., POD), its short development time, and its lower cost compared to that of traditional big satellites. CubeSats are not only very popular tools for educating students in the space research and engineering fields, but they also allow us to demonstrate challenging technologies and carry out science missions in space in a cheaper and faster way [4,5]. However, CubeSats often fail to achieve mission success. The vast majority of the failed CubeSat missions have been university-led projects [2]. Reliability improvements and prevention of poor workmanship are needed. Furthermore, reducing the development time remains a challenge [6].

The CubeSat standard has allowed for a huge expansion in the small satellite market. To date, more than 700 CubeSats have been launched into space [7,8]. The basic functions of standards (*reliability, information standards, compatibility, and variety reduction*) have positive effects on the technology market [9]. Therefore, using a standardized electrical interface for different CubeSat missions is one of the key ways to reduce costs and delivery times by taking advantage of modularity. The discussion about standardized bus interfaces is not a new topic, since several attempts at achieving them have already been made, such as the CubeSat Kit Bus [10] and the UNISEC interface [11].

The CubeSat Kit Bus interconnects all subsystem module boards via a stackable single connector that has 104 pins, the same physical connector as a PC/104 bus [12] but with a different pin assignment. This interface, first introduced by Pumpkin Inc., was later adopted by various CubeSat module developers. However, though a higher number of developers have been using this connector for their products, there are several disadvantages. A recent study reported issues with the PC/104 interface and concluded that a smaller connector with fewer pins than PC/104 is desirable [13]. Another issue with the PC/104 interface is its troublesome procedure when a middle board (of sandwich architecture) must be removed or replaced. All of the top or bottom boards must be removed to reach the targeted board.

The backplane interface board, also known as the UNISEC interface, introduced by the University of Würzburg in Germany, was demonstrated by a 1U CubeSat, UWE-3. The backplane utilizes a 50-pin connector, which is smaller than PC/104, into which each Printed Circuit Board (PCB) is inserted. The UWE-3 project proposes a set of pin assignments as a standard CubeSat electrical interface [11]. But, even if the assignment of a PCB differs from the ones in Ref. [11], there is no need for a jumper wire. We can absorb the difference of the pin assignment by rerouting the backplane pattern. The time required for assembly and disassembly is much shorter than that for a PC/104-based CubeSat, as there is very little need for wire harnessing and modification or replacement in both the flat-sat development and flight model integration stages is very easy to do. While introducing this interface standard, the UWE-3 designers tried to benefit more in terms of flexibility, robustness and efficiency [14].

Noting its advantage over the PC/104 style, we implemented the backplane approach for three CubeSat projects at Kyushu Institute of Technology (Kyutech). In those projects, nine CubeSats in total were built from scratch and sent into space over the past three years. BIRDS-1 [15] is a constellation of five 1U CubeSats with identical designs. They were deployed from the International Space Station (ISS) in July 2017. BIRDS-2 [16] is a constellation of three 1U CubeSats with identical designs. They were deployed from the ISS in August 2018. SPATIUM-I [17] is a 2U CubeSat. Internally, the PCBs are housed only in a 1U volume with another 1U having a mass dummy. SPATIUM-I was deployed from the ISS in October 2018. While implementing the backplane approach to those three projects, we discovered that issues with hardware changes on the backplane routing significantly decrease the advantages of this interface board.

The BIRDS-1 project implemented an architecture similar to that of UWE-3. The next two projects that followed, BIRDS-2 project and SPATIUM-I, tried to incorporate as much of the BIRDS-1 backplane design as possible to save on development time. However, the backplane interface boards needed to be changed for each new CubeSat project: UWE-3 to BIRDS-1, BIRDS-1 to BIRDS-2, and BIRDS-1 to SPATIUM-I. The flight models of the backplane interface boards for all four CubeSat projects are shown in Figs. 1 and 2. From UWE-3 to BIRDS-1, the changes occurred not only in the physical shape, the number of connectors, and their locations, but also in the routing of the PCB layout. Table 1 represents the results based on the number of interface connections that were changed from the UWE-3 backplane to that of BIRDS-1 to meet the BIRDS-1 requirements. The numbers in each cell of Table 1 (same for Table 2) represent the number of electrical connections between any two pins of any connector on the PCB board. The following reasons led to these changes.

Launcher requirements: The UWE-3 CubeSat was launched by a Dnepr rocket, while BIRDS-1 satellites were released from the ISS. Due to the specific requirements for deployment switches associated with the J-SSOD [18] that was used for the ISS release, BIRDS-1 needed the deployment switches changed.

Different module interfaces: The UWE-3 backplane was composed of seven 50-pin connectors. Each pin of those seven connectors was directly connected; for example, the no.1 pins of every connector were wired to each other. The BIRDS-1 backplane is made of five 50-pin connectors. The five connectors accept a Front Access Board (FAB), a motherboard carrying an Onboard Computer (OBC)/Electrical Power System (EPS)/Modem, a Communication Board (COM) carrying an Ultra High Frequency (UHF) transmitter, a Mission Board (MSN) as the main payload, and a Rear Access Board (RAB) carrying an antenna deployment mechanism. The BIRDS-1 main bus communication protocol was a Serial Peripheral Interface (SPI), which required more communication lines depending on the number of slave devices compared to I2C protocol. To maximize the effectiveness of the line usage, we cut some connections that do not necessarily need to go to every connector and used the empty pins for the new connection.

Accepting ambitious missions: BIRDS-1 had six missions in total. Four mission payloads were onboard. Due to space constraints, six microcontrollers were placed on the MSN. The MSN was not facing the outside. It was not even the closest board to the external solar panel. Therefore, the external access for programming/debugging of each microcontroller after integration required more connections on the backplane. An external access port was placed on the RAB. Since access pins and connections are used only on the ground, many connections were routed on the backplane between only those two boards. Similar to the previous example, connecting those pins to the connector's pins on other boards was useless and a waste of pins.

On the other hand, the BIRDS-2 and SPATIUM-I backplanes look very similar to that of BIRDS-1 (see Fig. 1), although the numbers and locations of 50-pin connectors are slightly different. This was mostly the result of a different size of payload being onboard. Apart from that, the interface connection on the backplane was changed. Table 2 lists the number of changes that were made to the BIRDS-1 backplane in the two projects. The critical problems that led to change routing on the backplane are as follows.

Discontinuity of COTS product: The radio transmitter and the modems of BIRDS-1 went out of production and disappeared from the market. Thus, BIRDS-2 could not take the heritage of the BIRDS-1 communication module, and a new design became necessary for the Communication Subsystem (COM). The new communication board could not accept the interface connection of the old communication board.

Specific design requirements: SPATIUM-I does not have an OBC system nor some of the simple controls performed by a microcontroller in the COM. The satellite has redundant power switches called "kill switches" to cut all the power of the satellite and permanently terminate radio transmission at the end of the satellite lifetime. Since, there is no OBC, the kill switch mounted on the EPS board needed to be controlled by the COM.

Those were the main problems that drove the changes from one project to another.

In addition, during each project, the interface definition changed several times; more specifically, there were three versions for BIRDS-1, two versions for BIRDS-2, and two versions for SPATIUM-I. Each time, changes were made to the pin assignment of the PCBs, and the backplane circuit needed rerouting. Even tiny changes require

complete reproduction of the backplane. Therefore, redesign of the backplane hardware resulted in a delay in the product delivery due to lengthy communications with the manufacturers. These changes were often caused by design issues that were discovered during the development and verification phases of the projects. Especially for the BIRDS-1 and BIRDS-2 projects, all of the members were students who did not have prior experience with satellite development. Inexperienced team members are often the cause of unforeseen design problems.

The purpose of the present study is to solve this hardware change issue by creating a standardized Software-Configurable Interface Board (SoftCIB) with a Complex Programmable Logic Device (CPLD) on the backplane board. The standardized backplane interface board will have the following advantages: It is (1) harnessless, (2) easy to assemble and disassemble, (3) compatible among different CubeSat projects, and (4) it has flexible routing. Figure 3 shows a conceptual diagram of how the SoftCIB will work. The SoftCIB can be used for various 1U CubeSat missions without changing any hardware, as only software changes by reprogramming the CPLD is required. For example, 1U CubeSat Project A and CubeSat Project B can use the same SoftCIB; only the interface connection must be reconfigured with CPLD software. Instead of designing and making new interface boards for new CubeSat projects, one can reuse the SoftCIB to cut the cost and development time involved in designing an interface board.

Different projects have different mission payloads and interface requirements. Thanks to the high interface flexibility of the SoftCIB, one can choose either the same or a different board for the bus system, such as an OBC, EPS, etc. We chose the backplane approach because it has more advantages than a PC/104, as we discussed. However, the idea of software rerouting can be used for any type of interface design that requires very high flexibility, if there is room for a CPLD.

This paper comprises four sections. The second section describes the SoftCIB design and development. The third section discusses the system level integration and space environment tests. The final section provides conclusions.

Table 1. Number of connection changes on BIRDS-1 backplane transformed from UWE-3

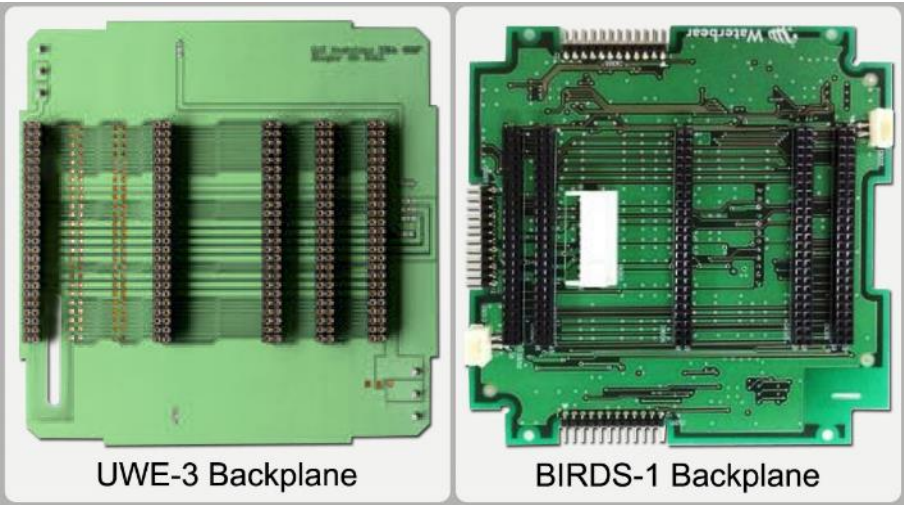| Connections	Not modified	Connections removed	New connections	Number of connections on BIRDS-1 backplane
Analog connection	32	54	10	42
Digital connection	50	135	17	67
Total	82	189	27	109

Figure 1. Backplanes of UWE-3 (left) and BIRDS-1 (right)

Table 2. Number of Changes on BIRDS-2 and SPATIUM-I backplane changed from BIRDS-1

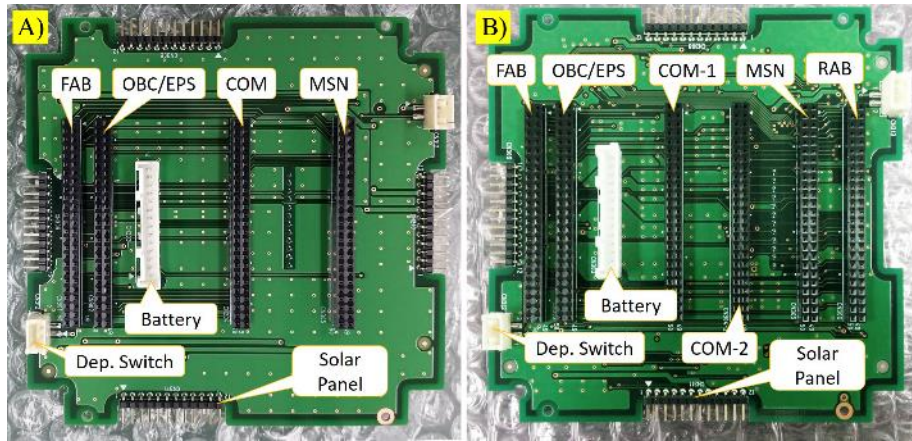| Connections	Not modified	Connections removed	New connections	Number of connections on backplane after modification
BIRDS-2 backplane	81	22	33	114
SPATIUM-I backplane	22	87	39	61

Figure 2. Backplanes of SPATIUM-I (A) and BIRDS-2 (B)

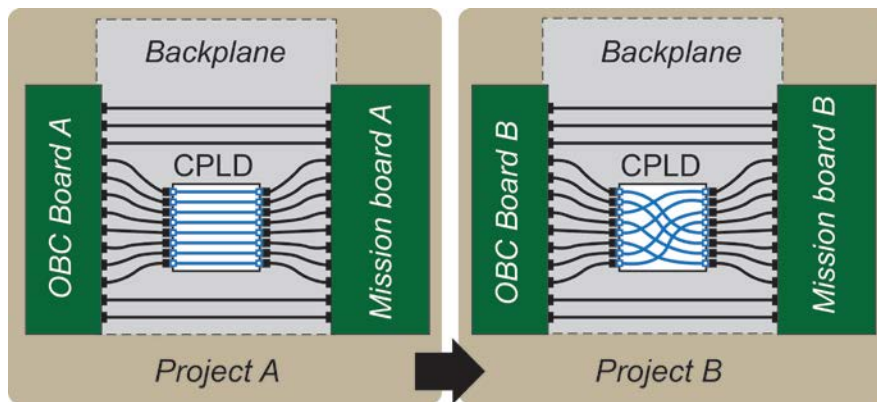## 2. Software-Configurable Interface Board (SoftCIB)

Design and development

Figure 3. Conceptual utilization of SoftCIB

The power consumption and physical space of a CPLD on the backplane board are the main concerns for the SoftCIB, due to the highly constrained size and power budget of 1U CubeSats. A few CubeSat missions have utilized CPLDs to date [19]–[20]. Unlike a Field-Programmable Gate Array (FPGA), CPLDs are not used for heavy processing or calculation onboard CubeSats. However, FPGAs used in 1U CubeSat missions often face the challenge of high power consumption [21]. In contrast, the SoftCIB is promising for lower power consumption than a FGPA. The former does not require any signal processing and calculations. As the acceptable maximum power consumption for the SoftCIB, we have set 100 mW as the target. An ispMACH® 4256ZE lattice device has been selected for the CPLD of the SoftCIB, considering its price, power consumption, temperature range, physical size and number of pins.

A simplified block diagram of the SoftCIB is shown in Fig. 4. The SoftCIB will include six general purpose units of 50-pin connectors and the same physical shape as found in the BIRDS-1 and BIRDS-2 six-layer backplane boards (*96.8 mm × 96.8 mm × 1.6 mm*). The 50-pin connectors will be used for the hub connections of various subsystems or module boards, such as FAB or EPS, OBC, COM, MSN, RAB or antenna board. Most data connections

excluding power lines will be handled by the CPLD on the SoftCIB, represented by the green arrows in Fig. 4. Therefore, the CPLD will act as electric routing for any data connection between different boards. All power lines are represented in dark blue in Fig. 4. From the experiences of our previous CubeSat missions, 5V, 3.3V and unregulated power lines are connected at fixed positions on the 50-pin connectors. Rerouting the power lines is undesirable. In addition, some data lines, represented in purple in Fig. 4, are directly routed between the 50-pin connectors. Those direct connections can be used for critical data connections such as those between OBC and COM boards to control the onboard transceiver.

Apart from the CPLD, the SoftCIB connects solar panels to an EPS board via printed wires on PCB. Similarly, the SoftCIB connects the deployment switches and the EPS via a harness. To maximize flexibility, there are two optional positions for the MSN, four optional positions for deployment switch connectors, and one temperature sensor to measure the temperature of the backplane board. Users can select the positions of the mission board and deployment switches, depending on their payload size or requirements.
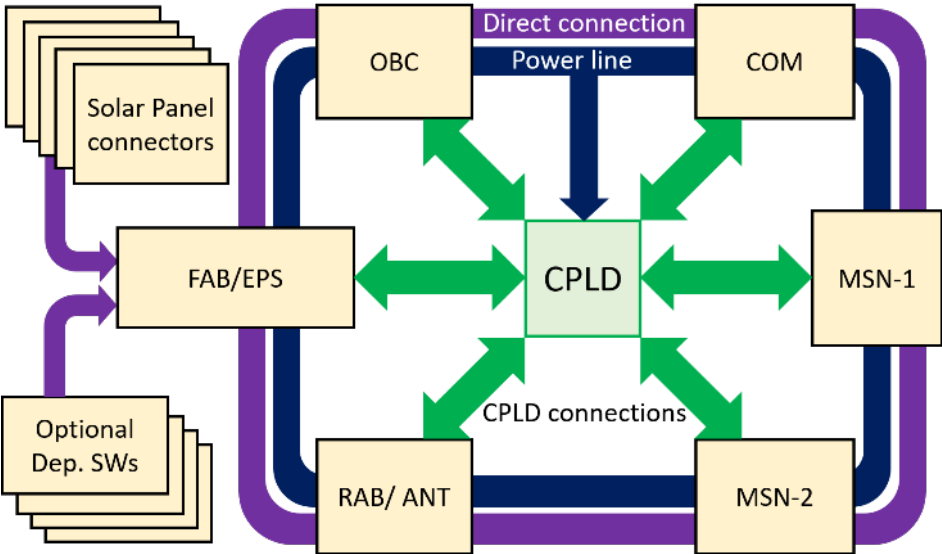

Figure 4. Block diagram of SoftCIB

Software

Compared to a FPGA, the SoftCIB will use much simpler software and algorithms. The basic function of the software is to define and allocate the input and output pins in VHDL, which is generally defined by the Interface Control Document (ICD) of satellite projects. In most cases, the CPLD acts as a simple digital follower circuit, since it is replacing the harness. For example, one of the output pins follows the logics of the assigned input pin.

Performance of the SoftCIB

First, to validate the concept of the SoftCIB, a functional test was conducted on the SoftCIB with a BreadBoard Module (BBM) and then with the prototype board. The OBC, MSN and FAB of the BIRDS-1 TableSat (same as a flight model of the BIRDS-1 satellite but configured for testing without external panels) was used to check the main function (interface connection) of the SoftCIB. The interface connectivity of the BBM is programmed to replace the BIRDS-1 backplane board. A DC power supply was used to supply the BBM instead of a battery. The TableSat boards with SoftCIB prototype board are shown in Fig. 5. Results of the functional test are shown in Table 3. All of the test results were good, represented as "○" in the Table. Functional test results for SPI communication and general H/L logic control between the MSN and microcontrollers on the OBC board, and UART communication between OBC microcontrollers and the PC through the CPLD board were good, and the total power consumption of the BBM board was 36 mW.

Figure 5. BIRDS-1 subsystem boards inserted on the prototype of SoftCIB

Table 3. Functional test results

Functions	BBM board	Prototype board
UART connection – OBC microcontroller and PC *(115200 bps)*	◯	◯
UART connection – COM microcontroller and PC *(115200 bps)*	◯	◯
UART connection – COM and OBC microcontrollers *(9600 bps)*	◯	◯
SPI connection – Flash Memory and OBC microcontroller *(1Mbps)*	◯	◯
SPI connection – Mission Board Memory and OBC microcontroller *(1Mbps)*	◯	◯
COM microcontroller reset through CPLD (H/L signal)	◯	◯
OBC microcontroller reset through CPLD (H/L signal)	◯	◯
Mission board and OBC board (H/L signal)	◯	◯

Table 4. Result of actual SPI communication test between Arduino and OV 5647 module

SPI communication speed	Success rate (With CPLD)	Success rate (without CPLD)
1 Mbps	100%	100%
2 Mbps	100%	100%
4 Mbps	95%	95%
8 Mbps	0%	50%

Second, the following two questions need to be answered. What is the highest speed of communication that a CPLD can handle? And how much signal delay might a CPLD produce? To answer these questions, we conducted two tests: A simple input and output test with a signal generator and oscilloscope, and actual SPI communication through the CPLD with a microcontroller and camera. For the simple input and output test, we used only two pins of the CPLD and programmed it as a follower, where the output follows the input logic. Then we gave the clock signal (3.3 V for the high level, 0 V for the low level) from the functional signal generator at the input, and measured the output by oscilloscope. Due to equipment availability, we increased the input signal frequency up to 40 MHz and observed the output. Figure 5 shows the input and output signal data measured on the oscilloscope. It must be noted that the signal shapes are not exactly the same and not square in Fig. 6 because the capacitance of the oscilloscope signal probes do not compensate well for high frequency measurement. However, it is clear and consistent that CPLD output follows the input without any interruption or signal loss and the signal delay is about 9 ns.
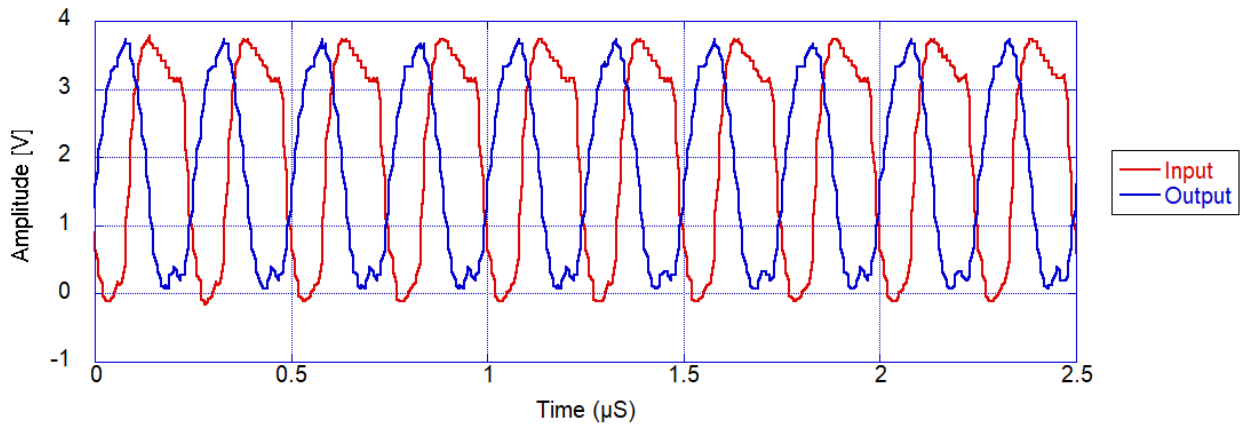
Figure 6. Comparison of the signals at the input and output of the CPLD pins

For the actual SPI communication test, an Arduino board was used as a master device and an OV5647 camera module was used as a slave SPI device. The SoftCIB board was programmed to replace four SPI lines between the Arduino board and the OV5647 camera. The other lines were not passed through the CPLD. The Arduino board sent the command to take the picture and download the image in JPG format via SPI from the camera module. The Arduino board was also connected to a PC via USB port to check the image quality. At each different SPI speed (1, 2, 4 and 8 Mbps), multiple images were taken and downloaded on the PC to determine the communication success rate. If there was any single bit error during the SPI communication, the JPG image could not be correctly reconstructed, thus we could easily recognize communication errors by looking at the images on the PC.

The test results are listed in Table 4. The results are compared with those from tests of a normal jumper cable connection which does not use a CPLD. The maximum SPI speed for the OV5647 camera module is 8 Mbps according to its datasheet. But the results show that SPI communication at 8 Mbps completely failed in the "with CPLD" test and partially failed in the "without CPLD" test. After the test, we investigated the cause of failure at 8 Mbps. To do so, we monitored all four inputs and the four CPLD outputs at the same time. Three SPI lines—CLK (Clock signal), CS (Chip Select), and MOSI (Master Output & Slave Input)—had no problems. The outputs gave exactly the same logic data as the inputs. However, the only output signal line of the camera module—MISO (Master Input & Slave Output)—had a problem. The signal at the input and output of the CPLD was slightly different, as shown in Fig. 7. The red color represents the CPLD input signal, i.e., the output signal of the camera module, and the blue color represents the CPLD output signal that was connected to the Arduino board. The falling edge of the input signal is not straight and has some noise. For the CPLD (Lattice ispMACH4000ZE family), the maximum value for the logic low signal is 0.8 V, which means that output signal cannot be logically low until the input signal goes below 0.8 V. Therefore, on the graph, the logically high output signal is longer than the input signal. This time difference becomes critical when one bit length is smaller or when the communication speed is higher. Furthermore, this creates bit errors and causes communication failure. The conclusion is that the output signal of the OV5647 camera module is not a good shape for a digital signal at a high frequency. The CPLD transferred the information, but failed to follow the falling edge of the signal.
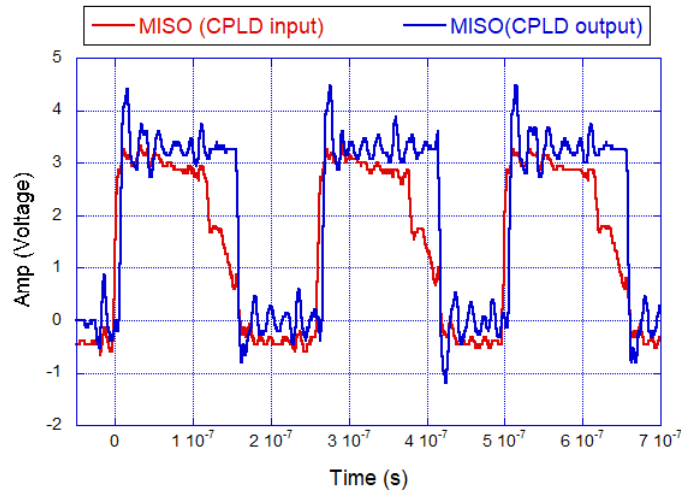
Figure 7. MISO signal at the input and output of the CPLD

Flexibility

The SoftCIB has 63 direct connections that are permanently routed on the PCB board and 46 flexible connections that can be configured by the CPLD. The term "connection" above refers to a single signal line between any two terminals (pins) of the 50-pin connector. In that case, 96 pins of the CPLD were used to route the connections (up to 46 connections) between 96 pins of the various 50-pin connectors. There are 31 power lines that are permanently routed on the PCB that handle all the power to the subsystems, including 3.3 V, 5 V, unregulated voltage, raw power of the battery, and the ground lines.

3. Integration and Environment Tests

Engineering model of BIRDS-3
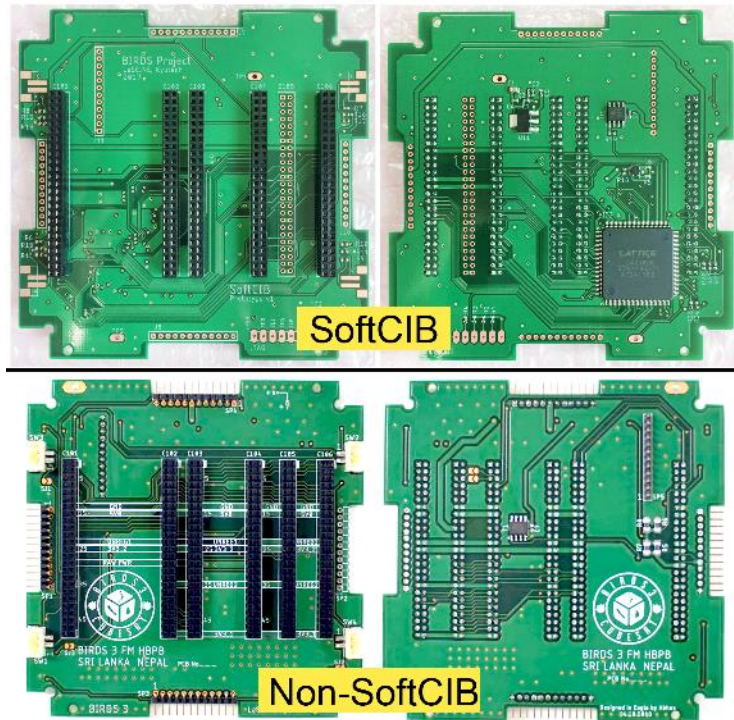
Figure 8. SoftCIB compared with Non-SoftCIB backplane board. Left, top view; right, bottom view

Demonstration of the SoftCIB is one of the BIRDS-3 satellite missions. The BIRDS-3 project is a constellation of three 1U CubeSats designed and developed by students who came from Sri Lanka, Nepal, Bhutan and Japan. The three satellites have identical designs, except for the backplane. Two satellites, Raavana-1 and NepaliSat-1, carry a normal backplane similar to the previous BIRDS projects [15,16], and the third one, Uguisu, carries the SoftCIB. Although there are two types of backplanes, the ICDs (e.g., pin assignment, physical dimensions, etc.) are identical. During development, every functional test of the satellites with different backplanes were compared. No failure or malfunction was detected. The Engineering Model (EM) of the BIRDS-3 satellite has been integrated and tested. The SoftCIB used for the BIRDS-3 EM is shown in Fig. 8. A photograph of the BIRDS-3 EM is shown in Fig. 9. The maximum power consumption of this SoftCIB was approximately 40 mW.

Figure 9. EM of BIRD-3 CubeSat

Vibration tests

A random test and a quasi-static acceleration test were done using HTV, SpaceX and Orbital Cygnus launch vehicle profiles at Qualification Test (QT) levels according to the JEM Payload Accommodation Handbook [18]. For the random vibration test, the overall Grms were 6.8, and the quasi-static acceleration was 22.6 (G) in all directions. There was no significant change in the natural frequency of the structure along all axes before and after the vibration tests. Functional tests were done before and after the vibration test, and no anomaly was found.

Thermal Vacuum test

The Thermal Vacuum Test (TVT) for the BIRDS-3 satellite EM was conducted using the small vacuum chamber at the Center for Nanosatellite Testing (CENT) at Kyutech. Figures 10 and 11 show the test setup and a photograph. A total of 20 thermocouples were attached to the satellite, including the internal components and external panels. A thermocouple was attached to the top of the CPLD as shown in Fig. 13. Pressure and temperature values were collected by DAQ and stored in the PC. The external power supply (PS1 in Fig. 10) was used to supply the heaters. A deployment switch was used as the ON/OFF control for the satellite. Instead of solar panels, an external power supply (PS2 in Fig. 10) was used as a power source to charge the battery. Functional tests were monitored and controlled by the Ground Station (GS) placed beside the thermal vacuum chamber, and a PC was connected to the satellite via a serial port. A RF cable and attenuator were connected to the GS and UHF transceiver of the satellite.

The chamber pressure was kept below 1 x 10^{-3} Pa during all conditions. In total, four thermal cycles were completed. The temperature of the external panel was controlled between -45 and 55ºC. Each cycle had a soak, both cold and hot, for 30 min. The functional test was then conducted. Figure 13 shows the actual temperature profile during the test. The fourth cycle was an experimental cycle where the cold soak was extended until the temperature of the battery reached -10ºC and a functional test was conducted.

The temperature of the external panels, for example, the –Y side (Minus_Y in Fig. 13) was the controlling temperature. The temperatures of the subsystems inside the satellite, including OBC, UHF transmitter, MSN 2, and the battery box are also shown in Fig. 13. The CPLD temperature is shown by the red color on the graph. The lowest temperature during the test on the CPLD was -42ºC, and the highest was 67ºC. All the functional tests were successful and no anomaly related to the CPLD was observed.
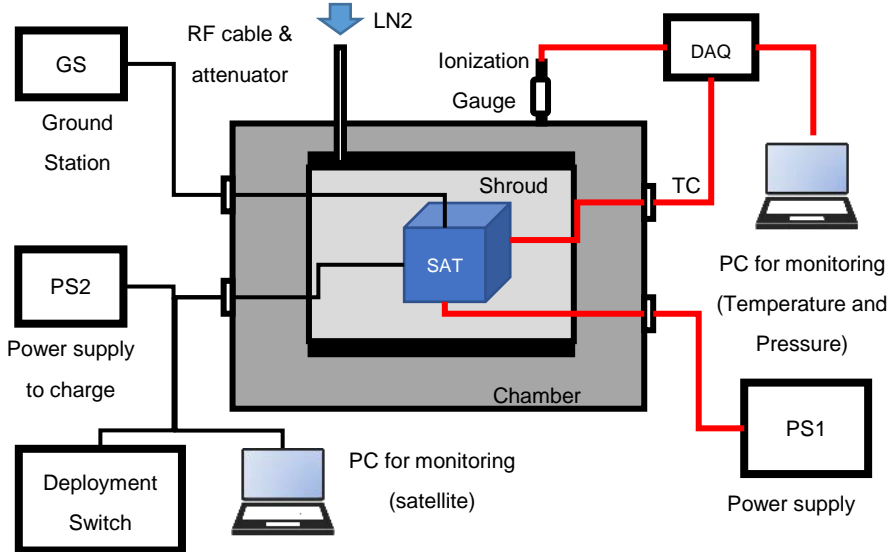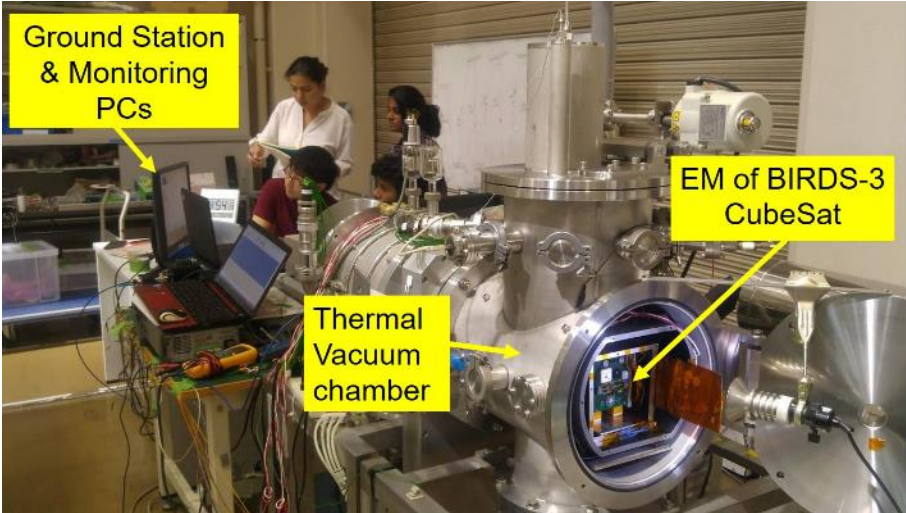

Figure 10. Schematic of TVT test setup

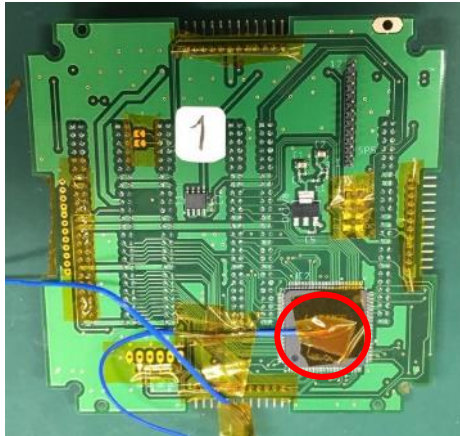Figure 11. General view of thermal vacuum test setup

11

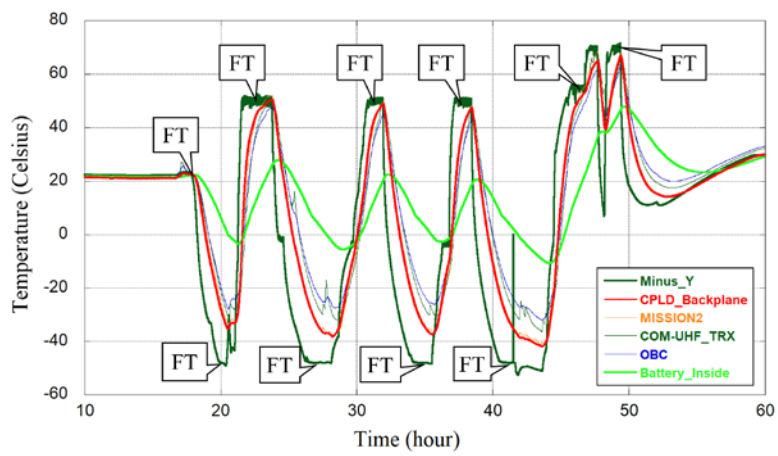Figure 12. Thermocouple placed on the CPLD of

the SoftCIB

Figure 13. Temperature profile of the TVT and points where Functional Tests (FT)

have performed

Table 5. Radiation dose at different samples at different distances

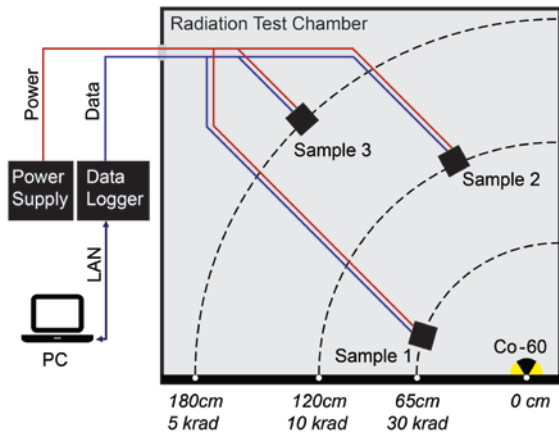| Test Article	Sample	Distance from the radiation source (*cm*)	Radiation dose (*krad*)
LC4256ZE7TN144I	1	65	30
	2	120	10
	3	180	5

Total Ionization Dose (TID) test
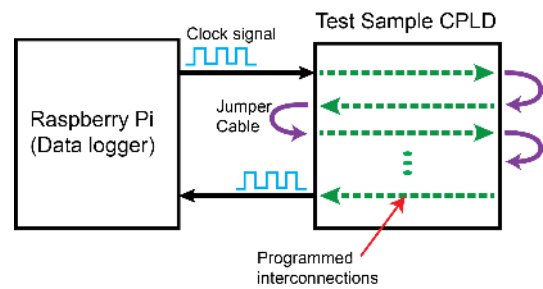

Figure 14. TID test configuration

Figure 15. Illustration of the test method for CPLD
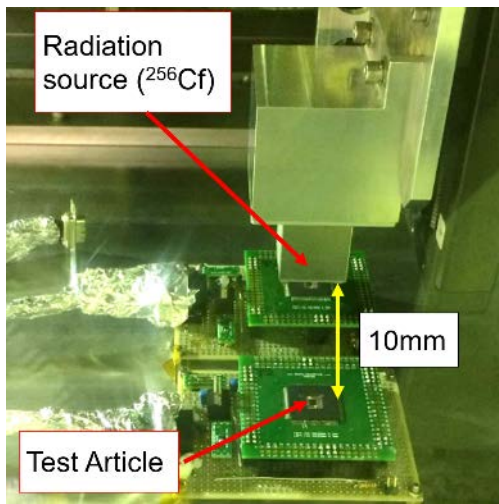
under radiation

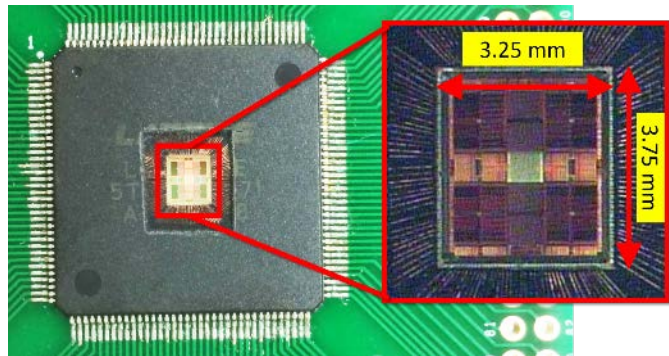Figure 16. Internal view of the SEE

test chamber

Figure 17. Photograph of CPLD under SEE test (the plastic

package was removed)

A total ionization dose test is necessary to verify that a Commercial Off-the-Shelf (COTS) CPLD can survive in a space radiation environment for a typical CubeSat lifetime. Only the CPLD, not the whole SoftCIB board, was tested, because the CPLD was the only critical electronic part on the SoftCIB. Three CPLD (part number: LC4256ZE7TN144I) samples were tested under three different dose conditions. The radiation source was Cobalt-60 (Co^{60}) and the radiation doses were estimated depending on the distance, as shown in Table 5. The test setup is shown in Fig. 14. The test method is illustrated in Fig. 15. The data logger, which was implemented by a Raspberry Pi computer, sent a 1-Hz clock signal to one of the CPLD pins and recorded the data from another pin. The CPLD is

programmed to replace multiple input-to-output connections. We used jumper cables to short all the CPLD outputs to another input (creating a chain) except the input/output to/from the data logger. Only the clock signal from the Raspberry Pi was transferred through the CPLD many times and came back to the Raspberry Pi. If one of the connections on the CPLD had a problem, the data would be lost. We regarded the test as a "failure" if data was lost. The test criterion was simply pass or fail. Performance degradation was not considered in this test.

No failure was recorded during the radiation test. All CPLDs operated normally up to a radiation dose level of 30 Krad, which was three times higher than the unit qualification test level defined in the ISO standards (ISO-19683:2017). The selected CPLD for the SoftCIB has the minimum radiation strength to survive in low Earth orbit within a typical CubeSat lifetime under the space radiation condition.

Radiation test for Single-Event Effects

It is also necessary to test the CPLD for Single-Event Effects (SEEs), since a SoftCIB has the risk of a single point of failure. We conducted the test using a radioisotope, californium-252 (^{252}Cf). A test facility with ^{252}Cf is more easily accessible and less expensive than a test facility with a particle accelerator [22]. The ^{252}Cf radiation test was almost the same as that in [22] and was carried out at the Kyoto University Research Reactor Institute. The test purposes were as follows: to detect any Single-Event Latch-up (SEL) by current measurement, to observe Single-Event Upset (SEU) by bit changes in the non-volatile memory, to confirm whether the CPLD can recover from a SEL by power reset, and to estimate the minimum SEL occurrence rate in orbit. An internal view of the test chamber is shown in Fig. 16. The radioactivity of the ^{252}Cf was 13.43 μCi, and 3.19 x 10^4 ions were released every second from a circle with a 10-mm radius. The estimated ion flux at CPLD (10 mm from the source) was 593 (ions/cm^2/s) with a 3% margin of error. A photograph of the CPLD during the test is shown in Fig. 17. The plastic packages of the CPLDs were removed because the heavy ions from ^{252}Cf could not penetrate it. A total of four CPLDs from the same lot were tested. The test article is a Lattice ispMACH4000ZE family CPLD with the part number LC4256ZE7TN144I. A schematic of the test setup is illustrated in Fig. 18. The CPLD was connected to a PC through the JTAG debugger to check the operation and to read and write all the configuration bits to the CPLD memory. We can identify a SEU by comparing the CPLD memory configuration data before and after radiation exposure. The current sensor was placed between the external DC power supply and the CPLD. The sensor's output was monitored on the oscilloscope and PC outside the chamber. The relay switch was used for the power reset which is controlled from the PC.

Table 6. Summary of observations. Time from each power-on to the subsequent SEL in the test

Target test Sample	Total exposure time (hour)	Nominal current (mA)	SEL current [mA]	Time until SEL occur (sec)
T1	3	4	17	3540
			34	1440
			19	1200
			32	1020
T2	1	4	34	252
			18	487
			34	282
			36	152
			18	568
T3	2	4	34	47
			19	894
			18	1638
			34	850
			34	414
			19	2378
T4	1	4	34	779

No SEU was observed during the test. The configuration bits in the memory were unchanged after radiation exposure. Ions from ^{252}Cf fission have a mean Linear Energy Transfer (LET) of 43 MeV/mg/cm^2 [23], which is

typically higher than the SEU threshold. During the test, a total of 1.82×10^6 ions bombarded the CPLD, but no SEU was observed. This means that the SEU cross-section is lower than 1.37×10^{-7} (1/ion/device). Because we cannot simply calculate SEU occurrences from a heavy particle test result, due to protons, the major radiation particle causing SEU in LEO, the fact that no SEU was observed is very promising.

Several SELs were observed. An example of current increases during the test is shown in Fig. 19. A SEL occurs at 92 s in the graph. The current consumption before the SEL was 4 mA, and increased to 34 mA after the SEL occurred. An approximately 30-mA current jump was observed in this example. The result of all the observations are summarized in Table 5. A histogram of the current jumps is shown in Fig. 20. The current jumps can be separated into two categories: approximately 30 mA and 14 mA. Mean time for SEL occurrence is 582 and 1529 s, respectively. We derive the minimum SEL occurrence rate in orbit by simply replacing our value with that from the H8 microcontroller test, which was previously done in [22]. The relationship of the cross-sections in the ^{252}Cf test and in orbit for the H8 microcontroller is used as a reference. The estimated number of SEL events in orbit per year is 0.0027 for the 14-mA current jump, and 0.0072 for the 30-mA current jump. Once a SEL has occurred, the current consumption jumps to higher values and remains in that state. After the power reset, the current returns to normal. Functional interrupts or malfunctions were not observed during the test.
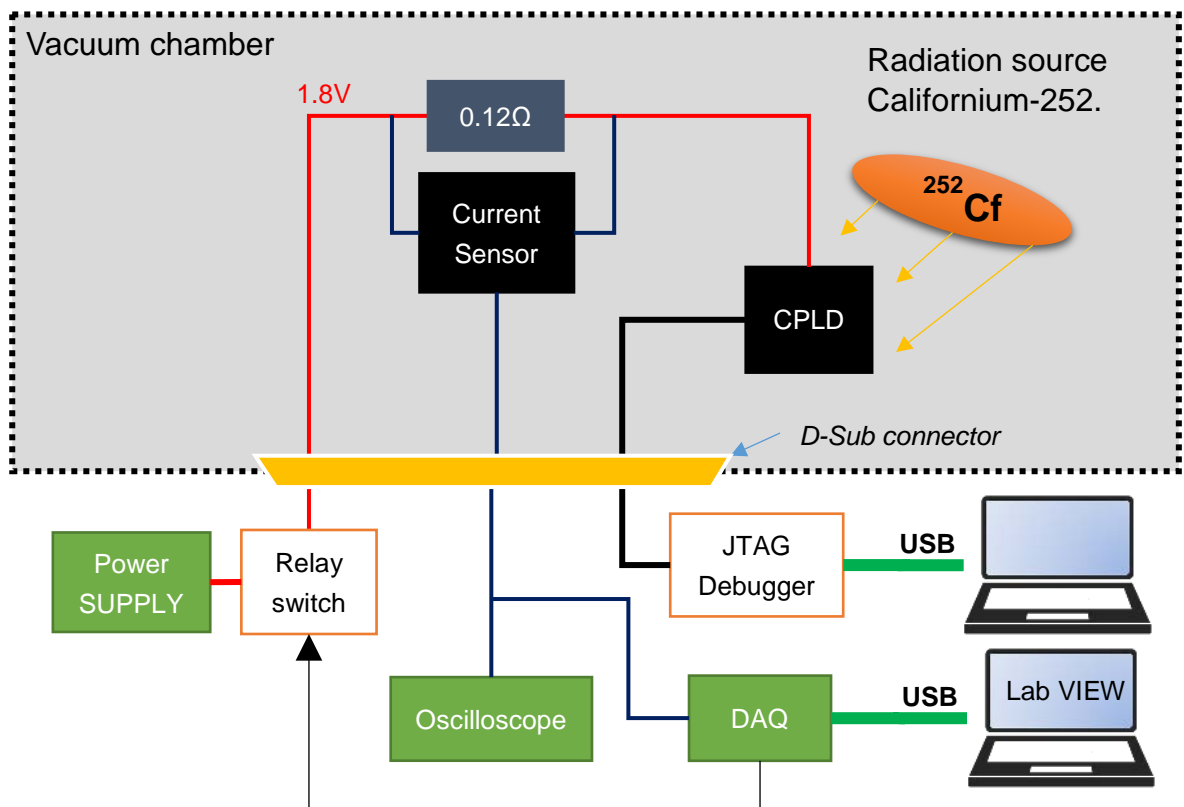

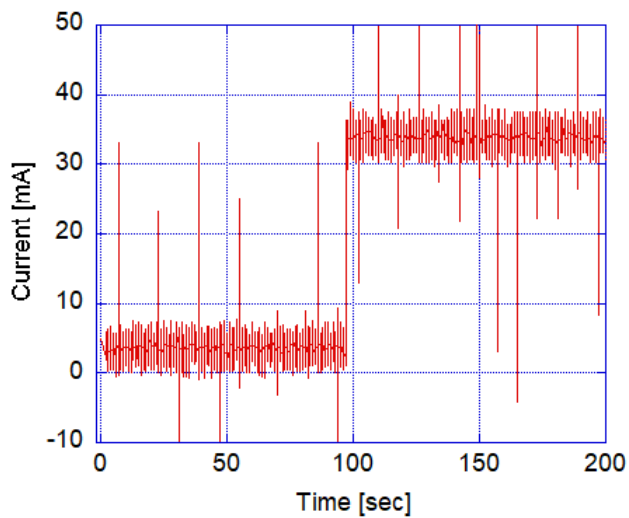Figure 18. Schematic of ^{252}Cf test setup

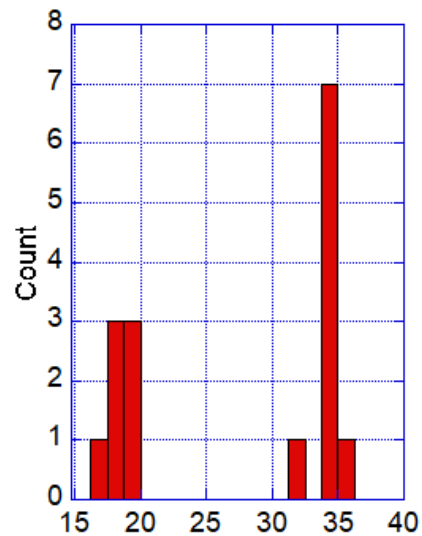Figure 19. Current jump observed during the test due to SEL occurrence

Figure 20. Histogram of the current increases due to SEL

Flight model test

The BIRDS-3 flight model with a SoftCIB is now complete. Figure 21 shows a photograph of the flight model. The configuration software has already been implemented on the CPLD. A functional test using the flight software is now being conducted. So far, after one week of operation, no anomaly has been observed. The satellite is planned to be delivered to JAXA in February 2019 along with the two other flight models with the hardwired backplanes. The three satellites will use the same flight software. They will be released from the ISS by the summer of 2019. The SoftCIB will be validated in orbit by comparing the in-orbit performance of the three satellites.

Figure 21. Flight Model CubeSats of BIRDS-3. CubeSat at the left top most corner is the Uguisu which carries SoftCIB.

Discussions

There are limitations for the use of a SoftCIB with CPLD. Firstly, the CPLD cannot transfer analog data. Only the digital signals can be handled through the CPLD. Secondly, implementing a bidirectional channel, whose input can be used as an output and vice-versa, requires one more pin of the CPLD to control the signal direction. Which means that I2C and similar protocols that uses one signal line for both directions are not possible to implement directly. However, it is possible to implement I2C by using an additional input and a support circuit that is defined by software inside the CPLD. Thirdly, Looking at similarity between UART and CAN buses, both uses RX and TX line separately, it is favorable to implement CAN bus through the SoftCIB. But the CAN and I2C buses have never been tested on SoftCIB. Therefore, an extensive test campaign to use I2C and CAN buses in the proposed switching architecture is necessary. Last but not least, SoftCIB still has the risk of a single point of failure if it is used for the mission critical data connection.

The SoftCIB concept is tested and utilized for 1U CubeSat size format. The idea can be applied to larger size CubeSats such as 2U, 3U, and 6U. From the authors' point of view, mechanical interfaces and shapes may be difficult to standardize the bigger backplane boards, because of freedom for bigger payload volume. Besides the standardization, however, we can still benefit from the harnessless design, and short time interface changing. Because, most of the student-developed CubeSat projects often faces issues of interface changing during the development, which we feel strongly from our own experience.

Some additional works may be necessary for the commercial CubeSat components already in the market to adapt the SoftCIB. The main works would be related to changing connectors on the PCB layout. This work may be difficult for some components where the PCB is already filled with many parts and the connector is not located at the edge of the PCB. CubeSat components with a PC/104 connector may find it easy to adopt to SoftCIB as the PC/104 connectors are located at the edge of PCB and easy to be replaced. For that case, however, the total number of the pins to be used shall be less than 50 including power lines.

Since, SoftCIB can reconfigure the interconnects by programming only, we can change the interconnects onboard even after the assembly, or even after the satellite is launched into orbit. Therefore, this approach may help in the future to create more software defined spacecraft, which can reconfigure the interconnections in orbit to accomplis the overall mission objectives.

4. Conclusions

Standardization of CubeSat interfaces contributes to shortened development and assembly times and reduced costs. A standardized interface will have high compatibility with various CubeSat projects whose missions differ. This research introduced the idea of a SoftCIB for a 1U CubeSat that does not require hardware rerouting even if the satellite mission requirements change. The concept was validated through testing via BBM and a prototype. The ispMACH® 4256ZE with 144 pin Thin Quad Flat Pack (TQFP) device was selected as the CPLD to reroute the connections among different PCBs. By implementing the CPLD, 42% of all data lines on the backplane board became programmable. Four programmable input/output and ten programmable input pins were not used, reserved for the future use. The power lines are fixed at particular positions of the 50-pin connectors.

The SoftCIB was selected for one of the BIRDS-3 CubeSat project missions and operation of the SoftCIB will be demonstrated in space. The EM of the BIRDS-3 CubeSat has been built and system level functional tests with the SoftCIB have been done. Space environment tests have been performed with the BIRDS-3 EM, such as the thermal vacuum test and vibration test. The overall result of the tests were positive; no abnormalities were found. The CPLD operated and started normally at lowest -42°C, and the highest 67°C temperatures. Thermal behavior, such as heat dissipations on the backplane board are not analyzed nor tested, since authors think that total power consumption of the CPLD device was very small. Furthermore, the CPLD passed a TID radiation test up to a dose of 30 Krad, confirming that a signal up to 4 Mbps could be transferred without problems. A radiation test using Californium-252 as the radiation source for SEE has been conducted. No SEU in the CPLD configuration memory was detected. We

learned that a CPLD can experience SEL. Two types of current jumps (14 and 30 mA) were observed during the test when a SEL occurred. It was proven that CPLD could recover from a SEL by power reset. In the present SoftCIB design, mitigation of SEL is a CPLD power reset by EPS. We did not implement an over-current protection (OCP) system for the CPLD alone on the SoftCIB since the current jumps are small. For the BIRDS-3 design, we have two 3.3-V outputs from the EPS. Both outputs have an OCP. One is used to power the CPLD, and the other is used to supply subsystems. The OBC has control signals to turn on/off those 3.3-V outputs, which means the BIRDS-3 OBC has the capability to reset the CPLD. Furthermore, BIRDS-3 has a regular time reset which resets the whole satellite every 24 h. The power consumption of the SoftCIB was 36 mW on average, which was at an acceptable level for even a power-limited 1U CubeSat.

Compared to the BIRDS-1, BIRDS-2 and SPATIUM-I projects, BIRDS-3 uses only one SoftCIB backplane hardware design for the flight model. The CPLD software was not changed during the development. More tests need to be performed to determine the maximum communication speed through the CPLD. The maximum signal speed confirmed was 4 Mbps and the CPLD did not function at 8 Mbps. Slave devices, that have been proven to communicate by SPI at high speeds, would have to be used for actual communication tests through the CPLD. Long-term reliability in the actual space environment will be tested onboard BIRDS-3, which will be released into orbit as early as the summer of 2019. Having a standardized software-configurable backplane board is an important step forward.

5. Acknowledgments

6. References

[1]   Cal Poly SLO, CubeSat Design Specification, REV 13, San Luis Obispo, CA, 2015.

[2]   M. Swartwout, The first one hundred CubeSats : A statistical look, J. Small Satell. 2 (2013) 213–233. http://www.jossonline.com/.

[3]   M. Cho, M. Hirokazu, F. Graziani, Introduction to lean satellite and ISO standard for lean satellite, RAST 2015 - Proc. 7th Int. Conf. Recent Adv. Sp. Technol. (2015) 789–792. doi:10.1109/RAST.2015.7208447.

[4]   J. Bouwmeester, J. Guo, Survey of worldwide pico- and nanosatellite missions, distributions and subsystem technology, Acta Astronaut. 67 (2010) 854–862. doi:10.1016/j.actaastro.2010.06.004.

[5]   K. Woellert, P. Ehrenfreund, A.J. Ricco, H. Hertzfeld, Cubesats: Cost-effective science and technology platforms for emerging and developing nations, Adv. Sp. Res. 47 (2011) 663–684. doi:10.1016/j.asr.2010.10.009.

[6]   A.K. Nervold, J. Berk, J. Straub, D. Whalen, A Pathway to Small Satellite Market Growth, Adv. Aerosp. Sci. Technol. 1 (2016) 14–20.

[7]   Erik Kulu, Nanosatellite and CubeSat Database, (n.d.). http://www.nanosats.eu/ (accessed December 16, 2017).

[8]   M. Swartwout, CubeSat Database, (n.d.). https://sites.google.com/a/slu.edu/swartwout/home/cubesat-

database (accessed December 17, 2017).

[9] G. Tassey, Standardization in technology-based markets, Res. Policy. 29 (2000) 587–602. doi:10.1016/S0048-7333(99)00091-8.

[10] A.E. Kalman, Overview of the CubeSat Kit, in: Inaug. Stanford - Cal Poly CubeSat Technol. Exch., 2004.

[11] UNISEC Europe, CubeSat Subsystem Interface Definition (Proposal), (2015) 0–10. http://unisec-europe.eu/wordpress/wp-content/uploads/CubeSat-Subsystem-Interface-Standard.pdf.

[12] PC/104 Embedded Consortium, PC / 104 Specification, (2008). https://pc104.org/hardware-specifications/pc104/.

[13] J. Bouwmeester, M. Langer, E. Gill, Survey on the implementation and reliability of CubeSat electrical bus interfaces, CEAS Sp. J. 9 (2017) 163–173. doi:10.1007/s12567-016-0138-0.

[14] S. Busch, Robust, Flexible and Efficient Design for Miniature Satellite Systems, Julius Maximilian University of Würzburg, Germany, 2016.

[15] Maisun Ibn Monowar, B.P. Team, M. Cho, Birds project: development and operation summary of a CubeSat constellation project, in: 68th Int. Astronaut. Congr. 24th IAA Symp. SMALL Satell. Mission. (B4), Small Satell. Mission. Glob. Tech. Sess., IAF, Adelaide, Australia, 2017.

[16] A.C. Salces, S.B.M. Zaki, S. Kim, H. Masui, M. Cho, Design, Development, Testing and On-orbit Performance Results of a Low-cost Store-and-Forward Payload Onboard a 1U CubeSat Constellation for Remote Data Collection Applications, in: 69th Int. Astronaut. Congr., IAF, Bremen, Germany, 2018.

[17] K. Aheieva, R. Rahmatillah, R. Ninagawa, H. Masui, T. Yamauchi, S. Kim, M. Cho, C.C. Lap, T.M. Siu, L. King, H. Holden, Space timing reference option for space applications provided by Space Precision Atomic-clock TIming Utility Mission satellite "SPATIUM," in: Jt. Conf. 31st ISTS, 26th ISSFD 8th NSAT, Matsuyama-Ehima, Japan, n.d.

[18] Japan Aerospace Exploration Agency (JAXA), JEM Payload Accommodation Handbook Small Satellite Deployment Interface Control Document, 2015. doi:JX-ESPC-101133-B.

[19] S. Deng, T. Meng, Z. Jin, Nonlinear programming control using differential aerodynamic drag for CubeSat formation flying, Front. Inf. Technol. Electron. Eng. 18 (2017) 867–881. doi:https://doi.org/10.1631/FITEE.1500493.

[20] S. Palo, X. Li, D. Gerhardt, D. Turner, R. Kohnert, V. Hoxie, S. Batiste, Conducting Science with a CubeSat: The Colorado Student Space Weather Experiment, 24th Annu. AIAA/USU Conf. Small Satell. (2012).

[21] S.S. Arnold, R. Nuzzaci, A. Gordon-Ross, Energy budgeting for CubeSats with an integrated FPGA, in: IEEE Aerosp. Conf. Proc., Big Sky, MT, USA, 2012. doi:10.1109/AERO.2012.6187240.

[22] T. Tomioka, Y. Okumura, H. Masui, K. Takamiya, M. Cho, Screening of nanosatellite microprocessors using californium single-event latch-up test results, Acta Astronaut. 126 (2016) 334–341. doi:10.1016/j.actaastro.2016.05.004.

[23] J.H. Stephen, T.K. Sanderson, D. Mapper, J. Farren, R. Harboe-Sorensen, L. Adams, A Comparison of Heavy Ion Sources Used in Cosmic Ray Simulation Studies of VLSI Circuits, IEEE Trans. Nucl. Sci. Vol. 31, Issue 6, Pp. 1069-1072. 31 (1984) 1069–1072. doi:10.1109/TNS.1984.4333457.