SSC19-WKIII-04

PyCubed: An Open-Source, Radiation-Tested CubeSat Platform Programmable Entirely in Python

Maximillian Holliday, Andrea Ramirez, Connor Settle, Tane Tatum, Debbie Senesky, Zachary Manchester
Stanford University
496 Lomita Mall, Stanford, CA 94035; (650) 723-2300
maholli@stanford.edu

ABSTRACT

The hardware and software pitfalls associated with satellite development have led to a nearly 60% failure rate among first-time CubeSat builders. We believe that a high-quality, community-driven CubeSat avionics platform would dramatically improve mission success rates. PyCubed is an open-source, radiation-tested CubeSat avionics platform that integrates power, computing, communication, and attitude determination and control functionality into a single low-cost PC104-compatible module programmable entirely in Python. PyCubed addresses many hardware-related failure modes through component and system-level radiation testing, in-depth design and qualification documentation, and flight heritage on a successful LEO mission. The challenge of flight software development is also mitigated through the use of the Python programming language via CircuitPython. Python enables dramatically simplified CubeSat software development, allowing students and researchers to quickly and reliably integrate new sensors, control algorithms, and payload hardware into their missions. All design files, including additional design justifications and reliability documentation, are available on the project's GitHub site.

INTRODUCTION

Small satellites have a rich heritage as educational and proof-of-concept engineering platforms. However, as cost-to-orbit decreases in response to improved launch opportunities, CubeSats are increasingly being used in critical commercial, scientific, educational, and technology-demonstration missions[1]. Unfortunately, the failure rates associated with CubeSats have remained consistently high[2].

The basic engineering associated with CubeSat development presents significant barriers to their use in education and scientific research. A set of high-quality, open-source CubeSat building blocks would lower these barriers. While the aerospace industry does not have a long history of open-source projects, community-driven hardware and software efforts have been hugely successful in related areas such as fixed-wing and multirotor "drones" and autonomous underwater vehicles[3,4]. The positive response to Planet's open-source release of their flight-tested radio at SmallSat 2018 is a testament to the community's need for these resources[5].

Although open-source CubeSat projects exist, a mature, well-tested hardware and software platform that addresses the cost, complexity, and reliability of these building blocks is still lacking. To address this need, we are developing PyCubed, a complete avionics stack that is designed for high reliability, radiation tolerance, and ease of use. PyCubed addresses many of the common pitfalls encountered in low-cost and/or first-time CubeSat building, supported by thorough documentation, qualification data, and flight heritage.

The paper proceeds as follows: In section II, we provide background information on radiation effects relevant to CubeSats. Section III introduces PyCubed and provides a detailed discussion of its hardware and software design. In Section IV, we present vibration and radiation test data, as well as flight data from PyCube's successful operation on the KickSat-2 mission. Figure 1 below illustrates the PyCubed flight board used onboard the 2018-2019 KickSat-2 mission.

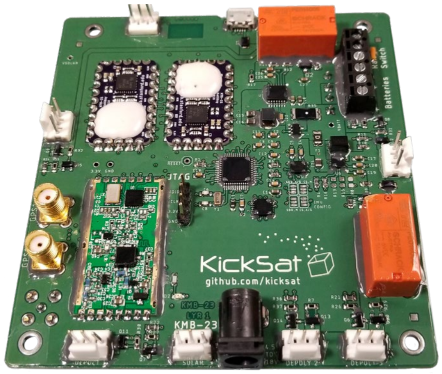

Figure 1: PyCubed Flight Board Prior to Integration into the KickSat-2 3U CubeSat.

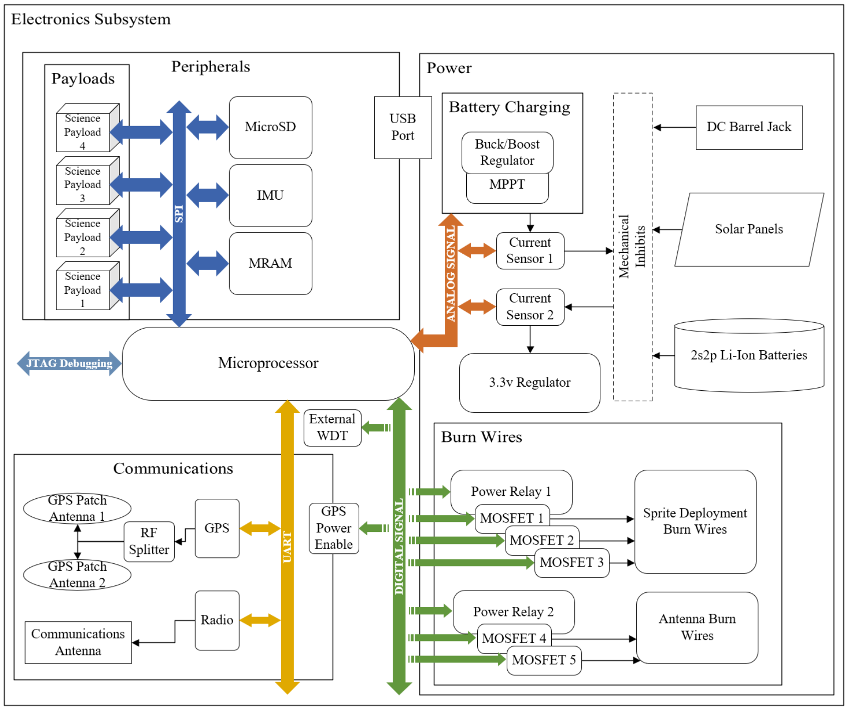

Figure 2: Functional block diagram of PyCubed Avionics Board.

BACKGROUND

There are many factors that determining how modern microelectronics behave in the harsh space environment, and radiation effects remain an active area of scientific research. This section provides the necessary background to understand the unique radiation mitigation strategies used in the design of PyCubed.

Ionizing radiation broadly describes a type of radiation able to transfer enough energy to displace electrons from atoms, and is the primary radiation concern for modern integrated circuits (ICs). The impact of displaced electrons on device function is difficult to predict, ranging from temporary to permanent damage, and is commonly determined experimentally[6]. Integrated circuits can be designed for improved radiation tolerance, but inherent design tradeoffs and market demand make it difficult for device manufacturers to prioritize this in development and fabrication.

The accumulation of radiation-induced damage inside a material is referred to as *Total Ionizing Dose* (TID) and is described in units of energy-per-mass (SI units of Gray, or Rad). It is helpful to picture TID in terms of a single transistor, where the cumulative degradation manifests as incremental shifts in threshold voltage and increased leakage current caused by electron/hole pair generation and trapping in the dielectric/interface regions. The TID threshold is the dose at which the device can no longer function within the manufacturer's specifications, and frequently determines the overall "lifetime" of a device.

In contrast to the gradual accumulation of TID, discrete high-energy particles can cause localized disruptions to electronic devices resulting in a *Single Event Effect* (SEE). The amount of energy a radiation particle can deposit per-unit-length into a material is called *linear energy transfer* (LET), and is a helpful metric for quantifying SEE potential. SEE can result in a range of behaviors, such as inverting the state of a logic transistor (bit flip), or the creation of an unwanted conductive path that can potentially draw enough current to locally melt regions of the device (single-event burnout). Note that radiation events that cause SEE also impart TID, but the number of particle events per unit area (known as *fluence*) with sufficient energy to cause SEE is typically far less than the sum of all ionizing particles interacting with the material.

Radiation type and dose rate can change drastically depending on location and source. The type of radiation encountered by a satellite in low-Earth orbit is very different from that found in geosynchronous orbit, which is again different from what a Mars mission might experience. When discussing radiation effects on satellite missions, it is important to state the target altitude and primary failure mode of interest (TID or SEE).

The ESA Space Environment Information System (SPENVIS) is a widely used online tool that can quickly estimate the expected radiation environment from specific mission input parameters[7].

PROJECT OVERVIEW

PyCubed is an open source hardware and software platform for CubeSats. The functional block diagram shown in Fig. 2 is implemented on a single PC104-sized PCB using low-cost, widely available COTS hardware, resulting in the following capabilities:

- Command and control of the spacecraft
- Energy harvesting and power management
- Telecommunication
- Data collection and storage
- Payload interfacing
- Fail-safe deployment mechanisms

The design and component selections are supported with flight data, vibration and radiation testing, and thorough documentation on the project's public GitHub[8].

Originating as a high-reliability design, PyCubed served as the motherboard for the successful KickSat-2 LEO CubeSat mission conducted from November 2018 to March 2019. As depicted in Fig. 1, the single board performed all necessary spacecraft functions, executing the mission objective of deploying over 100 femtosatellites after 5 months in orbit. The design integrates the demanding ISS safety and responsible-orbital-debris requirements inherent to the KickSat-2 mission while maintaining open design documentation and unique code simplicity.

The operating envelope for KickSat-2 defined the environmental requirements of PyCubed, with deployment at 300 km and a temperature profile ranging from -20C to +100C.

The usefulness of a CubeSat as a science or research tool, as opposed to an engineering design exercise, is just as dependent on accessible software as it is reliable hardware. PyCubed addresses the need for software usability through implementation of MicroPython, an open-source real-time Python interpreter running on the primary microprocessor. Python is used by students, scientists, and engineers because of its simplicity and flexibility, making it the fastest-growing programming language[5]. MicroPython allows Python code to run directly on the sort of low-power microcontrollers commonly used in CubeSats.

HARDWARE DESIGN

The electronic design of PyCubed utilizes a modified form of the Careful Commercial-off-the-Shelf (COTS) technique outlined by Sinclair and Dyer [9]. The intent of Careful COTS is to provide an alternative to historically rigorous radiation-hardened satellite design that better fits the needs of modern small satellite projects. Advantages of Careful COTS include significant capability, cost, and schedule improvements at the expense of increased mission failure risk. The robust PyCubed design exemplifies key aspects of Careful COTS, including a thorough understanding of the targeted radiation environment, data-driven component selection, and supplemental radiation testing. Figure 3 illustrates the core sections of the board.

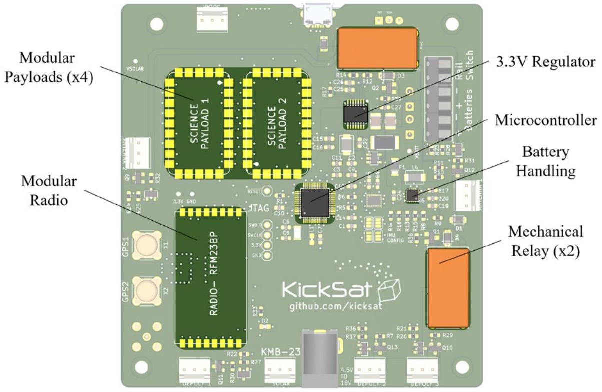

Figure 3: Illustration of PyCubed PCB with Key Hardware Components Highlighted.

Environmental modeling and component selection

SPENVIS was used to generate high-level orbital parameters and subsequent radiation models for the 300 km KickSat-2 orbit. Figure 4 summarizes the total simulated dose over 1 year at an orbit of 300 km for a range of aluminum enclosure thicknesses. Further analysis was performed using raw AE9/AP9 data in conjunction with Geant4, COMSOL, and VisualTCAD tools to model the anticipated radiation effects. A conservative total ionizing dose (TID) threshold of 10 krad (in silicon) was established for the mission using the method described by Sinclair and Dyer. Components with known TID behavior that met the threshold criteria were selected using the following online radiation testing databases:

- IEEE Radiation Effects Data Workshop Proceedings[10]
- NASA GSFC Radiation Database[11]

- European Space Components Information Exchange System[12]
- NASA JPL RAD Archive[13]

Avionics design and component selection is an iterative process when using the Careful COTS method. If a component did not have radiation data available, an appropriate substitute was selected. When a substitute with similar functionality did not exist, design-dependence on the component was re-evaluated. For crucial components that lacked TID data and sufficient flight heritage, TID radiation testing was performed to qualify the component as discussed in the Hardware Testing section. Mission-critical components are discussed below to illustrate this iterative design process.

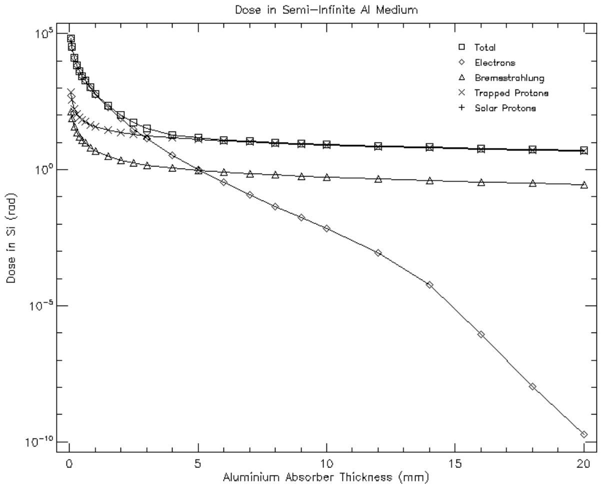

Figure 4: Simulated Dose Accumulation as a Function of CubeSat Shielding Thickness (1 yr. Mission Duration at 300 km Altitude).

Power

The TPS542XX family of DC-DC converters from Texas Instruments has been evaluated for TID and SEE tolerance by multiple researchers[14,15]. Specifically, Cochran et al. reported a TID tolerance of 15 to 20 krad, and Allen et al. reported no destructive SEL events occurring on devices biased at 10V or less and under a variety of temperature conditions. The efficiency of the TPS542XX converters, coupled with known radiation behavior and configurable output voltages makes it a good candidate for LEO missions.

Microprocessor

A Microchip ATSAMDXX family of microprocessor was selected to operate the PyCubed board. Although there is no published literature on the radiation tolerance of this device, reasonable inferences can be made from Cochran et al., 2008 TID work on a "complex 65nm CMOS microprocessor" as the basis for the TID threshold of an ARM M4 core processor[16]. Using the MIL-STD883 1019.8 test method, Cochran et al. reports no degradation detected up to the maximum tested level of 1 Mrad. Furthermore, the ATSAMDXX product family has flight heritage onboard the Brown Space Engineering EQUiSat 1U CubeSat[17].

The specific ATSAMD51 part was chosen because of its high performance 32-bit ARM Cortex-M4 architecture, low power consumption, flexible pin configuration, hardware floating point math support, on-board USB handling, and wide accessibility. However, the lack of direct radiation testing data for the ATSAMD51 justified performing TID qualification testing as discussed in the Hardware Testing section.

Watchdog

Crucial to the single-processor design of PyCubed is an external watchdog timer (WDT) that supplements the internal WDT of the ATSAMD51. The MAX70X series of microprocessor supervisors have an extensive radiation test history covering TID, SEE, and SEL scenarios from several researchers[18,19]. As reported by Kenna et al, the MAX70X devices have a TID threshold of roughly 11 krad under worst-case biasing conditions. Additionally, Allen et al. reports a SEL threshold of 72 LET at 25C and about 68 LET at 85C. Both TID and SEE values are well within the required mission parameters.

Data Storage

The MR25H40 is a 4MB non-volatile magnetoresistive memory device (MRAM) from Everspin Technologies. The fundamental operation of magnetoresistive memory is inherently more tolerant to TID effects. Cochran et al. reported a TID tolerance of 90–100 krad during their 2008 work at Goddard Space Flight Center Radiation Effects Facility[20]. Additionally, O'Bryan et al. was unable to observe upsets in the device using 89 MeV and 189 MeV protons during their 2008 work at the Indiana University Cyclotron Facility[13].

The high radiation tolerance of the magnetic memory on PyCubed is used for both reliable non-volatile memory code and data storage, as well as RAM for the ATSAMD51 microprocessor during operation of the real-time Python interpreter.

Large-scale data storage is also available on PyCubed in the form of a micro-SD card socket. The TID tolerance of commercial SD cards varies greatly depending on the charge-storage mechanism and NVM controller used by the card. In general, flash memory fabricated for single-level cells has greater TID thresholds than devices built with multi-level cells. Supported by findings from Kingsbury et al., the Delkin U1000-3 family of SD cards was selected for the KickSat-2 mission[22].

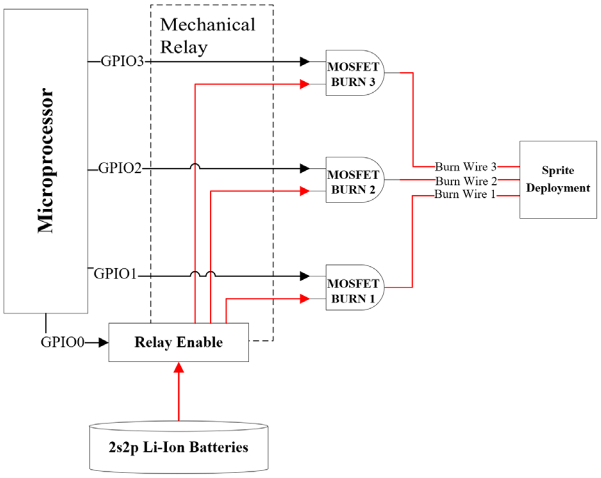

Figure 5: Radiation Tolerant Burn Wire Block Diagram.

Deployment

KickSat-2 had to satisfy strict deployment requirements to prevent undue risk to the ISS. The resulting burn wire scheme implemented on PyCubed is illustrated in Fig. 5. The MOSFETs and power relays controlling the deployment circuits were carefully selected to have well-documented electrical behavior in radiation environments. Incremental degradation due to ionizing radiation generally manifests as a *decrease* in gate threshold voltage for NMOS devices and an *increase* in threshold voltage for PMOS[6].

Both the IRLML5103 and IRLML2803 were evaluated by O'Bryan et al. in 2001 for TID tolerance using the NSWC Crain facilities[23]. Testing with the MIL-STD883 1019.8 procedure, including biased and unbiased conditions, the data shows the IRLML5103 has a TID tolerance greater than 35 krad, and the IRLML2803 greater than 30 krad. Both values far exceed the projected mission dose and are further confirmed from the experimental testing results shown in the Hardware Results section below.

Device failure due to single event effects (SEE) such as burnout and latch-up are significantly reduced for the burn-wire design because the circuit sits dormant (shorted to ground) by a mechanical relay during nearly the entire mission. By protecting the burn-wire circuit with the mechanical relay shown in Fig. 5, there is nominally no bias across the MOSFET devices, thereby preventing an ionizing particle from creating parasitic effects within the channel/bulk regions and causing permeant damage.

The PE014006 power relay is an inherently mechanical device that can only be engaged when enough current is driven across the input terminals to actuate a mechanical switch. It therefore is not susceptible to TID degradation. The construction of the device employs a large isolation distance which also protects the relay from SEE-actuated events.

Payload Modularity

The modularity of the PyCubed PCB design allows the project to maintain flexibility for researchers while allowing the CubeSat community to share, iterate, and improve it over time.

KickSat-2 successfully utilized a HopeRF RFM23BP radio transceiver on a surface-mount mezzanine board for telecommunications, as highlighted in Fig. 3. However, this radio location on the PyCubed PCB is equipped with power and standard UART, SPI, and I2C communication that can be used for many different commercial or custom CubeSat radio solutions. This design philosophy was also applied to the four science payload locations, which can support a variety of sensors or student-designed experiments. The payload locations accommodate surface-mount mezzanine boards and have a standard 0.1-inch pad spacing allowing for chassis-mounted payloads needing cables.

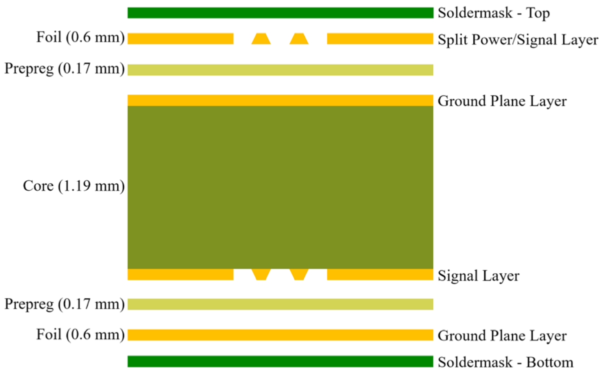

Figure 6: Four-Layer PCB Stack Up Design.

Reliability Best Practices

The reliability of the PyCubed PCB was enhanced through system-level reduction of thermal mechanical fatigue. Interconnect failure is a common environmentally induced failure mode for PCBs that manifests as solder joint fractures and plated-through hole (PTH) separation resulting from thermal or vibration-induced fatigue[24]. Thermal mechanical fatigue can be significantly reduced by selecting a PCB laminate material with lower coefficient of thermal expansion (CTE) in the out-of-plane axis as compared to traditional FR4 laminates. A reduced CTE results in less solder joint and PTH strain causing less thermal mechanical fatigue. For example, the FR408HR laminate material made by Isola Group is a widely available PCB material that reports a 30% improvement in Z-axis expansion[25]. Additionally, larger diameter plated-through holes were used to geometrically reduce PTH strain.

PyCubed is designed for 50-ohm controlled impedance using the 4-layer FR408HR laminate stack-up depicted in Fig. 6 with generous via diameters and trace widths. This stack-up reduce EMI emission and absorption, while maintaining signal integrity and thermal robustness. The result is a cost-effective board design that reduces manufacturing risk by not pushing the limits of modern PCB fabrication techniques and can selectively be made using more expensive laminate materials when necessary.

Interconnect failure was further reduced through component selection. For example, when possible, PyCubed uses quad-flat packages (QFP) over quad-flat non-leaded (QFN) packages because of the well-documented reliability issues associated with QFN devices[26]. The gullwing leads used on QFP packages have the added benefit of easier soldering/rework over the castellations found on QFN packages.

SOFTWARE IMPLEMENTATION

PyCubed is designed to leverage an embedded Python interpreter to provide a robust CubeSat software architecture that is easy to learn, faster to program, and simpler to troubleshoot than traditional embedded C and C++ approaches[27]. CircuitPython, a subset of MicroPython developed by Adafruit and the open source community[28], was chosen because of its emphasis on usability for beginner programmers.

Software Highlights

- Native USB support with serial port debugging, no extra drivers or other software required.
- CubeSat mounts as a "USB flash drive" with software and libraries stored as plain-text files on the drive, no compilation required.
- Simple Python programming syntax capable of accessing low-level hardware peripherals such as I2C, SPI, UART, and CAN bus.
- Open-source architecture with robust fault tolerance and error handling
- Capable of over-the-air software updates.

MicroPython's implementation as a real-time Python interpreter on embedded hardware makes its inherent state-machine architecture and code execution well-suited for high-reliability applications (Fig. 4).

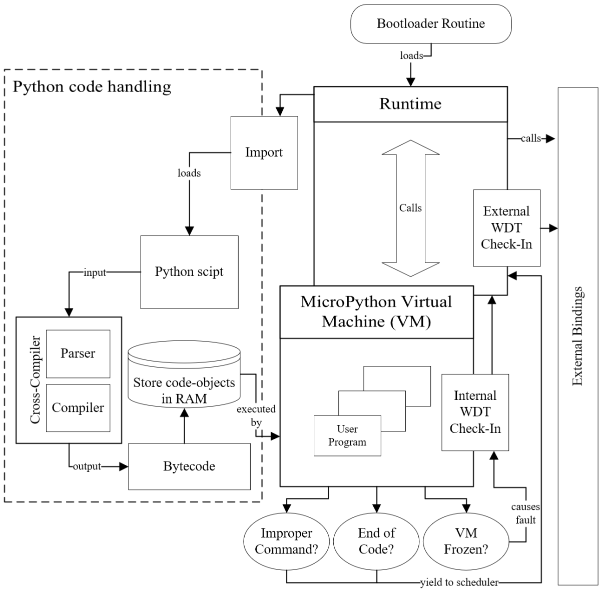

Figure 7: MicroPython Architecture Flowchart Used on PyCubed Operations.

Micropython Architecture

Simply put, MicroPython is low-level Python interpreter, written in C, running on the microcontroller. The interpreter parses Python code in plain-text files and operates the hardware accordingly. As illustrated in Fig. 7, this manifests as a runtime environment with a python "virtual machine" (VM) that is entirely transparent to the end user. Upon boot, the runtime uses the ARM processor to import python text files, converting the instructions to byte code and caching them in RAM partitioned on the radiation-tolerant external magnetic memory. Errors encountered in the interpreted Python code, whether user or radiation induced, do not cause the processor to halt. Instead, the runtime is instructed to proceed to the next python file (which could be a failsafe routine, for example) in the event of encountering un-executable code. The result is a software approach with robust fault tolerance not present in many educational or low-cost CubeSats.

The fault tolerance of the PyCubed software is further enhanced by integration of the external WDT. The WDT monitors the runtime environment, which is pre-programmed to periodically communicate with the WDT on sub-second intervals. Missing the communication window is an indication of a runtime fault, and the WDT will respond by power-cycling the processor with an external toggle. Upon restarting, the microprocessor is able to detect the un-planned reboot and act accordingly (by default, only recording the event in a log file).

MicroPython has been considered for space applications before, and has been ported to run on the LEON family of radiation-hardened processors[29,30]. The CircuitPython interpreter used on PyCubed is an adaptation of MicroPython to run on Microchip's ATSAMD microprocessor line. With simple syntax and extensive library resources, CircuitPython is targeted towards beginner programmers and provides the all of the tools necessary to control and use complex microelectronic systems.

Functional Software Example

There is no setup required to begin prototyping with the hardware and software on the PyCubed board. Using a USB cable to connect the board to a computer, the device will mount as storage media and present as a USB "flash drive," regardless of the operating system (Windows/Mac/Linux). All software is in the form of text files with the extension ".py" located on the drive. Upon opening the file "main.py" in a text editor, the user is able to alter or add code that will immediately begin running upon saving the file. There are no drivers, development environments, or compilers required. Figure 8 illustrates the simple and intuitive nature of programming PyCubed in Python. In less than 15 lines of code, the board is initialized, connection to multiple sensors is established over I2C, the radio is setup over SPI, and an operational loop is started.

```
import time, board, busio
import cpc
import adafruit_lsm9ds1 as IMU

# I2C connection:
i2c = busio.I2C(board.SCL, board.SDA)
sensor1 = IMU.LSM9DS1_I2C(i2c, address=0x40)
sensor2 = IMU.LSM9DS1_I2C(i2c, address=0x45)

# SPI connection:
spi = busio.SPI(board.SCK,
                MOSI=board.MOSI,
                MISO=board.MISO)
radio = CC1101(spi, freq=434400000, "666A")

radio.setupRX()
radio.setupCheck()

# Loop
while True:
 if 'take measurement' in radio.reciveData():
   [print(x, y, z) for x,y,z in sensor1.acceleration]
 time.sleep(1)
```

Figure 8: Functional Software Example.

HARDWARE TESTING RESULTS

Facilitated by the KickSat-2 mission, PyCubed passed all qualification testing necessary for launch on the NG-10 mission, including the random vibration test profile shown in Figure 9.

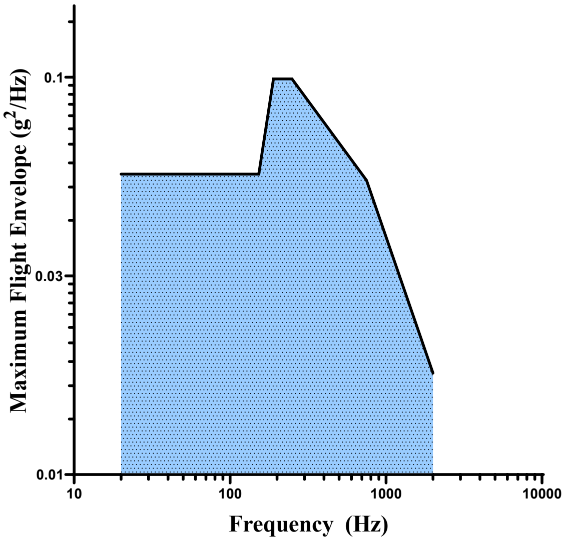

Figure 9: Power Spectral Density Plot of Vibration Qualification Requirements.

For mission-critical components that lacked sufficient radiation data, TID threshold limits were experimentally determined per MIL-STD883 1019.8 using a Shepherd Mark 1 gamma irradiator with a resulting photon energy of 662 keV. A summary of radiation findings is presented in Table 1. Although the specifics of radiation testing VLSI components such as modern microcontrollers are beyond the scope of this work, the summarized preliminary findings show a high confidence in these components surviving calculated mission dose levels determined for the LEO environment.

Table 1: Experimentally Measured TID Failure Threshold Values of Commercial Components.

Component Function	Part Number	TID Response
N-Channel MOSFET	IRLML5103	Within spec beyond 35 krad
P-Channel MOSFET	IRLML2803	Within spec beyond 35 krad
Microcontroller	ATSAMD51G	Flash memory issues starting at 16 krad, power and logic functional beyond 35 krad
Analog to Digital Converter	ADS124S08	Within spec until 19 krad

Flight Data

PyCubed successfully completed its first LEO mission in May 2019 after operating the KickSat-2 spacecraft for the duration of its 5-month mission. Mission highlights include deployment of over 100 femtosatellites and collection of magnetic field data from radiation-tolerant gallium nitride hall-effect sensors fabricated by the Stanford XLab group[31]. Figure 5 shows on-board battery voltage and dosimeter data sampled over the duration of the mission. The battery voltage data further validates the MPPT solar charging circuit, which reports nearly full each time the battery pack is measured, despite the spacecraft transmitting a status message every 60 seconds from its radio.

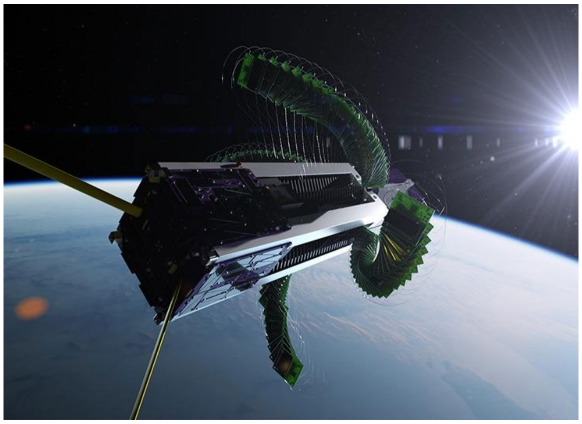

Figure 10: Artist Rendering of KickSat-2 Mothership, Commanded by the PyCubed Avionics Board as It Deploys Over 100 Femtosatellites.

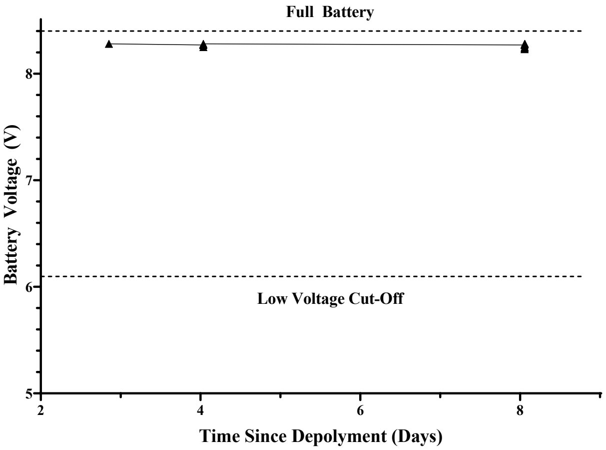

Figure 11: On-Orbit Battery Voltages Reported by PyCubed During KickSat-2 Mission.

Dosimeter data was measured from a Tyndall TY003 "RadFET" located on a mezzanine board soldered to the PyCubed motherboard. The threshold voltage of the RadFET was measured over time and normalized to remove temperature effects in order to calculate an accumulated dose inside the spacecraft. Figure 7b shows the calculated dose as a function of time starting with integration of KickSat-2 into the Cygnus P-Pod on Oct 23th, 2018, 4 months before deployment at 375 km and 7 months before re-entry. The dosimeter data before and after deployment allows for calculation of 34 Rad of accumulated dose while stored in the P-Pod and docked to the ISS for 90 days.

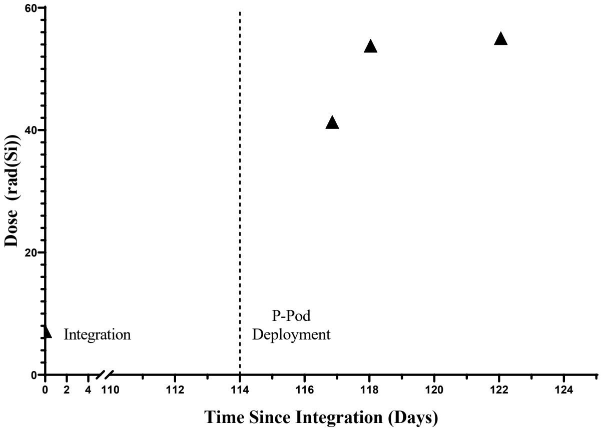

Figure 12: Dosimeter Data Collected by PyCubed During KickSat-2 Mission.

CONCLUSIONS AND FUTURE WORK

The open-source CubeSat hardware and software platform PyCubed has been developed to mediate the difficulties encountered by developers of low-cost CubeSats. Thorough design methodology and PCB fabrication best-practices have been utilized, and both radiation and vibration testing have been performed to validate the performance of the hardware. Additionally, PyCubed has achieved flight heritage on the recent KickSat-2 mission, further demonstrating its reliability.

All design files, including additional design justifications and reliability documentation are available on the project's GitHub repository[8]. Readers are encouraged to "fork" the design and build their own PyCubed boards, allowing them to modify, document, and ultimately contribute their own knowledge and experience to the open-source project.

PyCubed will serve as a foundation for future CubeSats developed in Stanford's Robotic Exploration Lab, and will be continuously improved. Additionally, beginning in Fall 2019, the PyCubed hardware will be utilized in Stanford's Aeronautics and Astronautics undergraduate capstone courses.

Acknowledgments

The authors would like to thank BJ Jaroux, Anh Nguyen, Sergio R. Santa Maria, and Sharmila

Bhattacharya of NASA Ames Research Center. This work was partially supported by NASA under agreement number 80NSSC18K1673.

References

1. Swartwout, M. CubeSats: Toys, Tools or Debris Cloud? (2019).
2. Swartwout, M. CubeSat Mission Success: Are We Getting Better? (2019).
3. Meier, L., Honegger, D. & Pollefeys, M. PX4: A node-based multithreaded open source robotics framework for deeply embedded platforms. in *2015 IEEE International Conference on Robotics and Automation (ICRA)* 6235–6240 (2015).
4. Lam, B. A Mini Sub Made From Cheap Parts Could Change Underwater Exploration. *New York Times* (2012).
5. Iraci, G. & Gnam, C. An Open Source Radio for Low Cost Small Satellite Ranging. *AIAAUSU Conf. Small Satell.* (2018).
6. *Defects in microelectronic materials and devices*. (CRC Press, 2009).
7. Royal Belgian Institute for Space Aeronomy & European Space Agency. SPENVIS - Space Environment, Effects, and Education System. *The Space Environment Information System.* https://www.spenvis.oma.be/regulation.php. (Online, 2019).
8. PyCubed, *GitHub*. https://github.com/pycubed. (Online, 2019).
9. Sinclair, D. & Dyer, J. Radiation Effects and COTS Parts in SmallSats. 12 (2014).
10. Radiation Effects Data, *IEEE Workshop*. https://ieeexplore.ieee.org/xpl/conhome.jsp?punumber=1000609. (Online, 2019).
11. GSFC Radiation Data Base. *NASA GSFC.* https://radhome.gsfc.nasa.gov/radhome/raddatabase/raddatabase.html. (Online, 2019).
12. ESCIES (European Space Components Information Exchange System). [Online]. https://escies.org/webdocument/showArticle?id=228&groupid=6 (Online, 2019).
13. JPL RAD Archive. *NASA JPL.* https://radcentral.jpl.nasa.gov/index.cfm?fuseaction=ShowRadFrame. (Online, 2019).
14. O'Bryan, M. V. *et al.* Compendium of Recent Single Event Effects Results for Candidate Spacecraft Electronics for NASA. in *2008 IEEE Radiation Effects Data Workshop* 11–20 (2008).
15. Cochran, D. J. *et al.* Total Ionizing Dose and Displacement Damage Compendium of Candidate Spacecraft Electronics for NASA. in *2009 IEEE Radiation Effects Data Workshop* 25–31 (2009).
16. Cochran, D. J. *et al.* Total Ionizing Dose and Displacement Damage Compendium of Candidate Spacecraft Electronics for NASA. in *2010 IEEE Radiation Effects Data Workshop* 8–8 (IEEE, 2010).
17. EQUiSat Deployment Announcement | Brown Space Engineering - Space For The People. *Brown Space Engineering* (2018).
18. Allen, G. R. Compendium of Test Results of Single Event Effects Conducted by the Jet Propulsion Laboratory. in *2008 IEEE Radiation Effects Data Workshop* 21–30 (2008).
19. Kenna, A. J., Rax, B. G., Thorbourn, D. O., Harris, R. D. & McClure, S. S. Compendium of Recent Total Ionizing Dose Test Results Conducted by the Jet Propulsion Laboratory from 2003 through 2009. in *2009 IEEE Radiation Effects Data Workshop* 32–38 (2009).
20. Cochran, D. J. *et al.* Compendium of Recent Total Ionizing Dose Results for Candidate Spacecraft Electronics for NASA. in *2008 IEEE Radiation Effects Data Workshop* 5–10 (2008).
21. O'Bryan, M. V. *et al.* NASA Goddard Space Flight Center's Compendium of Recent Single Event Effects Results. in *2018 IEEE Nuclear & Space Radiation Effects Conference (NSREC 2018)* 1–8 (IEEE, 2018).
22. Kingsbury, R. *et al.* TID Tolerance of Popular CubeSat Components. in *2013 IEEE Radiation Effects Data Workshop (REDW)* 1–4 (IEEE, 2013).
23. O'Bryan, M. V. *et al.* Current single event effects and radiation damage results for candidate spacecraft electronics. in *IEEE Radiation Effects Data Workshop* 82–105 (IEEE, 2002).
24. Vianco, P. and Holliday, M., Use of the TurboSiP© software to predict the long-term reliability of solder joints on photovoltaic systems. in *2013 IEEE 39th Photovoltaic Specialists Conference (PVSC)* 3029–3032 (2013).
25. Isola Group. FR408HR - High Performance Laminate and Prepreg. Datasheet. (2019).
26. Kong, R., Tulkoff, C. & Hillman, C. The Reliability Challenges of QFN Packaging. (2017)
27. Miller, J. A. Promoting Computer Literacy Through Programming Python. 301 (2019)
28. CircuitPython, *Adafruit*. https://circuitpython.org/. (Online, 2019).
29. George, D. P. Porting of MicroPython to LEON Platforms. 21 (2014)
30. Laroche, T., Denis, P., Parisis, P. & George, D. MicroPython Virtual Machine for On Board Control Procedures. 8 (2016)
31. Alpert, H. S. *et al.* Effect of Geometry on Sensitivity and Offset of AlGaN/GaN and InAlN/GaN Hall-Effect Sensors. *IEEE Sens. J.* **19**, 3640–3646 (2019).