Open-Source Flight Computer Platform for CubeSats

Charles Lindsay, Ethan Sit
Massachusetts Institute of Technology
77 Massachusetts Ave. MIT 37-350 Cambridge, MA. 02139
cmll@mit.edu, ethansit@mit.edu

Faculty Advisor: Kerri Cahoy
Massachusetts Institute of Technology

ABSTRACT

BeaverCube is a 3U CubeSat in development by the Space Systems Development capstone class and STAR Lab at MIT. The satellite will serve as a testing platform for an electrospray thruster developed by Accion Systems and host a payload with a mvBlueFox visual imaging camera and two FLIR Boson infrared cameras. The BeaverCube Flight Computer (BFC) is designed to serve as the power and data interface for the entire CubeSat bus, and incorporates a Raspberry Pi compute module as the central processor. Key design objectives for the BFC include a novel fault detection, isolation and response (FDIR) system to improve CubeSat reliability, the capability to interface with devices and power on the bus through standardized connectors, and a high data rate imaging payload. The outcome of this design is a flight computer able to recover autonomously (without ground intervention) from software faults or non-destructive hardware failures within 90 seconds. Two onboard computers and four data storage cards provide mission continuity if a destructive event occurs. The BFC supports several common data protocols including USB 2.0 for rapid payload data transfer, which allows the BFC to be readily integrated into many CubeSat systems. The BFC open-source repository will be made public following testing on the first revision prototype so this board design is available for future CubeSat missions.

Introduction

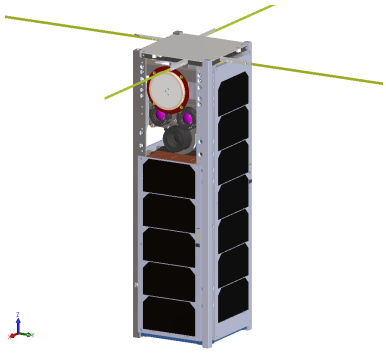


Figure 1: BeaverCube Flight Model

The central objective of the BFC was to develop an open source flight computer (FC) board capable of interfacing with several CubeSat components for a complete optical imaging system. BeaverCube requires a FC to process data from other subsystem components. The FC also houses the software controlling the satellite.

Constraints on the BFC include a board size of 90 mm x 96 mm, which limits the number of onboard computers and peripherals. The FC has to support a wide range of subsystem communication protocols spread across multiple boards, including I^2C, RS-485, TTL UART, SPI and USB. All subsystem boards, including the BFC, are stacked vertically in Beavercube, with the exception of the imaging payload and radios. Boards are fitted with a combination of standardized CubeSat connectors such as the PC/104 and additional unique connectors, both of which are able to interface with the FC board.

The BFC includes two Raspberry Pi computers which run mutually exclusively, with the secondary flight computer only running in the case of primary flight computer failure. Previously developed CubeSat missions including DeMi and AAReST also employed a dual Raspberry Pi flight computer architecture, although both computers operated simultaneously on the DeMi FC.[1,2] In addition, a programmable logic circuit was used on the AAReST FC to select which Raspberry Pi could communicate with external devices, whereas the BFC employs a

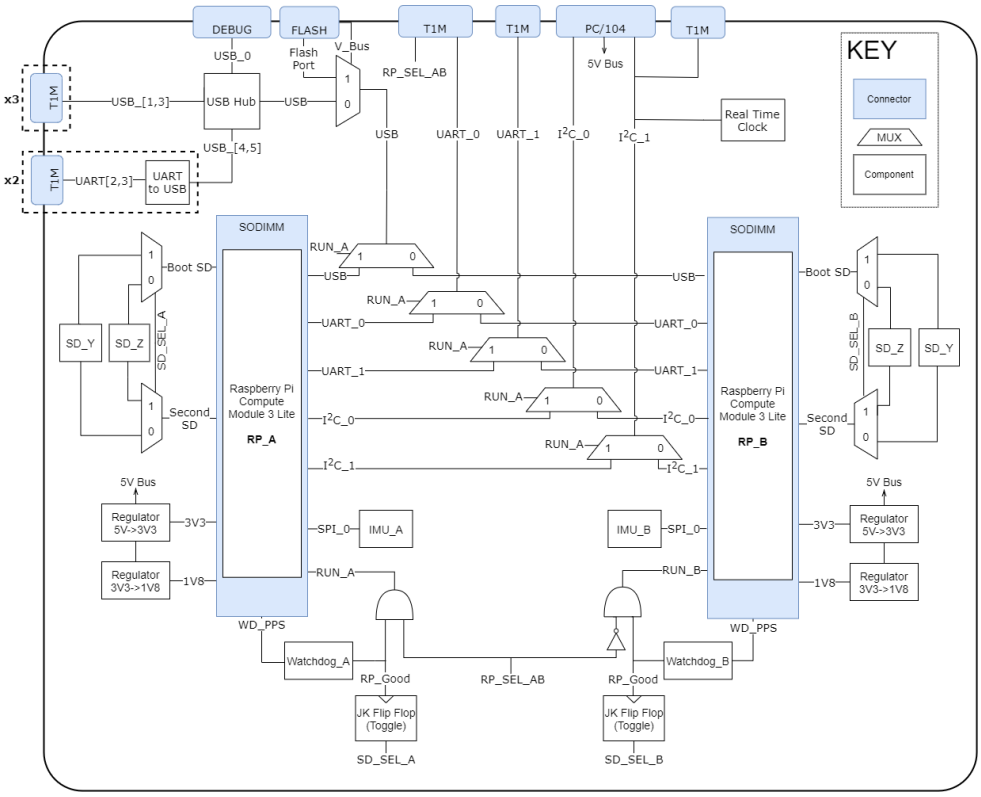


Figure 2: BFC Functionality and Interfaces, Rev 1.0, 4/24/2020

custom logic circuit configuration implemented entirely in hardware.[2]

Commercially available flight computers are typically built with a single ARM core running Linux or less processor-intensive operating systems, a single SD card mount for large data storage, and a PC/104 and several low profile connectors for external devices. Data interfaces include an array of I^2C, UART, SPI and USB protocol lines to allow for adaption to most CubeSat systems.

Key Design Features

Signal Assignment

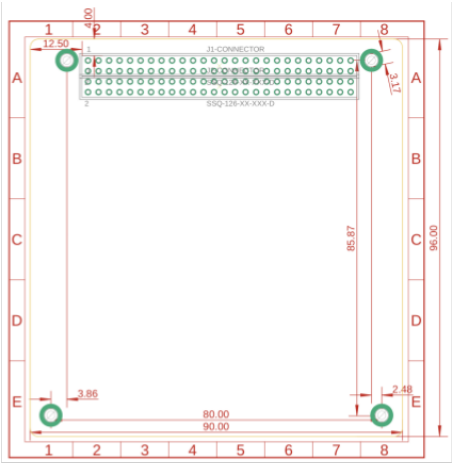
To maintain signal integrity and increase system reliability, great consideration was placed into signal assignment and communication protocols. The flight computer is constrained by available general-purpose input/output (GPIO) pins and available pins on the PC/104 connectors.

The basic structure of command and data handeling (CDH) interfaces can be broken down into three components. The center of CDH operations is the Raspberry Pi Compute Module 3 Lite (CM3L). While the CM3L itself has 200 pins in total, signals can only be transmitted between CDH and other subsystems using its 45 GPIO pins. This is because GPIO pins are unique from normal pins in that they can be configured for a variety of data transfer protocols.

There are multiple paths that signals can take from the CM3L to other subsystems' components. One option is using the PC/104 bus located on the CDH board. The PC/104 bus is a set of two 52 pin headers named H1 and H2 as shown below. Using a standard header allows boards to be stacked on top of each other, saving space inside the CubeSat. However, this means other subsystems' components are located off the CDH board and thus must be specified in this interface control document (ICD). In general, subsystem components are powered using the PC/104 bus.

In some cases, subsystems share the same data transfer protocol and this allows non-power signals to be multiplexed or connected to a hub instead of the PC/104 bus. The use of multiplexers and hubs still require the use of GPIO pins, but multiple com-

ponents are now allowed to communicate with the board using the same bus/pins.



Depiction of common template layout. Numbering of pins from upper left; H1 at top of image, H2 immediately below it.

Figure 3: BFC PC/104 Header Arrangement

Fault Detection, Isolation and Recovery

A major focus for the BFC was developing a novel CubeSat FDIR system in order to increase mission lifespan and flight success rate. The design goal for this system was to reduce the failure rate of the most critical components onboard the BFC, including the processor, data storage, radios and inertial measurement units (IMUs) used for position control. The design also focused on swift autonomous recovery of critical systems with as little ground intervention as possible. This was accomplished by implementing redundant logic circuits and sensors using inexpensive commercial-off-the-shelf (COTS) components instead of radiation hardened components.

Non-destructive errors due to cosmic radiation will inevitably cause the BFC SD card image to corrupt, and single event latchups (SEL) may cause components such as the voltage regulators to fail and cut power to a processor.[3] In addition, each Raspberry Pi contains 45 GPIO pins with transistors susceptible to SEL. It is necessary to isolate data lines from each processor that connect to the same devices in order to avoid interference in the case of a SEL, such as a I^2C select pin failing low and locking up that data bus.

The BFC incorporated several features of the payload computer recently used on the 6U DeMi CubeSat.[1] This includes use of a Raspberry Pi computer with additional logic circuits that will autonomously toggle the boot SD card flash memory in the event of a single event upset (SEU) in memory or manually toggle via a telemetry command from ground.

Several concerns from DeMi testing were addressed in the development of the BFC. While DeMi had dual computers for redundancy, both were simultaneously powered and on separate circuit boards with a UART communication link between boards.[1] In addition, several data lines between subsystems and the flight computer board were shared between Raspberry Pi computers, which could result in a data link failure in the event that a SEL causes one Raspberry Pi to fail with a GPIO pin held in an undetermined state. This may be unrecoverable and result in the failed data link interfering with CubeSat operations through the end of the mission. The BFC incorporates dual Raspberry Pi computers for fault tolerance, but implements mutually exclusive processor run logic. This reduces the flight computer power consumption and reduces the system failure rate, as the disabled flight computer will be less susceptible to latchups.

Power Management

The DeMi flight computer experienced brownouts during boot in testing, as the bus power between the flight computer and electrical power system (EPS) was routed through several connectors on the vertical hardware stack causing voltage sag when current demand changed. Our design was also concerned with inrush current tripping the EPS power bus during board power up.

A point of load conversion on the BFC was selected to decouple the Raspberry Pis from the EPS and avoid brownouts. Dual switching voltage regulators (3.3 V and 1.8 V) were chosen over linear regulators to decrease thermal power dissipation. The soft start feature was configured for 1 ms to prevent inrush current from tripping the EPS power bus.

Imaging Peripherals

A unique design challenge presented by the Beavercube mission requirements was supporting a large number of imaging peripherals. There is one visual cameras and two infrared cameras that needed to be synchronized. It was also desirable to minimize GPIO pins required by the Raspberry Pi to capture and store images.

The approach taken to address this was to multiplex signals through a USB hub. This reduced the required number of GPIO pins, provided a fast interface (480 Mbps for USB 2.0 protocol), and allowed for simultaneous image capture.

Feature Implementation

FDIR Analysis

Critical systems that are duplicated on the BFC include Raspberry Pi computers, voltage regulators and IMUs. The BFC is at least one fault tolerant to any of these components failing. Nominal operation for the BFC requires at least one Raspberry Pi computer to be receiving power with one SD card mounted that contains a non-corrupted image.

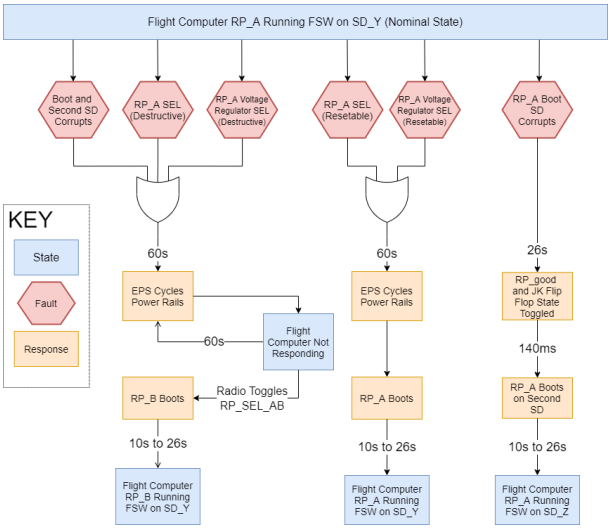


Figure 4: FDIR system overview. The transition time between states is indicated. Time to recover from a non-responsive flight computer is dependent on detection by a ground station and subsequent command uplink to toggle RP_SEL_AB. Assumes the flight computer powers on with RP_A booted to SD_Y

The BFC is triple fault tolerant to flash memory failure, with a single fault recoverable autonomously and more than one fault recoverable via a ground command. If a watchdog signal implemented in flight software fails to trigger the Raspberry Pi run logic (PiRL), the SD connected to the Raspberry Pi is swapped and the Raspberry Pi reboots on the new SD. The faulted SD is mounted to the Raspberry Pi secondary SD interface which allows it to be re-flashed to correct the image. In addition, stored data on the corrupted SD may be read for fault diagnosis and data recovery.

The BFC is single fault tolerant to a Raspberry Pi or voltage regulator failure during nominal operation, with a non-destructive SEL recoverable autonomously and a destructive failure recoverable via a ground command. The PiRL ensures that only a single Raspberry Pi can be running at any time. The EPS is always monitoring the I^2C interface to the active computer and if the Raspberry Pi or input voltage regulator faults, then the EPS will cycle the power rail which clears any SEL.[3] If the failure is destructive then a command from ground is required to toggle the Raspberry Pi that is running.

The computer run logic can only be triggered by a watchdog timer connected to the respective Raspberry Pi and a dedicated control signal tied directly to a GPIO pin on the primary radio's microcontroller. The primary radio onboard BeaverCube will be programmed to trigger this control line only via a preset command received from ground.

Flight Computer Design Process and Specifications

Eagle CAD was used for schematic capture and board layout. In order to decrease electromagnetic interference (EMI) and coupling between high frequency USB signals and other data links, a six layer board was selected. This allows for ground planes adjacent to all signal planes, and a dedicated plane for power distribution. A minimum microstrip trace width and spacing of 6 mils was used. Our design is completely open-source.

BFC Specifications	
Processor	BCM2837B0 1.40 GHz
Flash Memory	Micro-SD Card
RAM	1GB LPDDR2
USB Ports	1x Debug
	1x Flash
T1M Connectors	3x USB 2.0
	1x TTL UART -> USB hub
	1x RS-485 UART ->USB hub
	2x TTL UART -> RP GPIO
	1x I2C
PC/104 Connector	2x I2C
	5V, 3.3V, 12V to BFC
Integrated Sensors	1x Real Time Clock
	2x ICM-20948 IMU
Onboard PDU	2x 3.3V Regulator
	2x 1.8V Regulator
Redundancy	2x SD Card per RP
	1x Regulator per RP
	1x IMU per RP

Figure 5: Flight Computer Features

A spreadsheet was created to keep track of signals going into and out of the board. This spreadsheet

was used to maintain an ICD, which specified signal name, communication protocol used, and destination on or off the board. Use of the ICD allowed for signals to be properly routed in Eagle.

Future Work

We will be creating an open-source repository to host the most current version of the flight computer following testing and verification. As of the date of submission of this paper, the first revision of the BFC has yet to be tested. As the redundant Raspberry Pi system architecture for the DeMi CubeSat has been verified, key testing objects for the BFC include verifying the PiRL allows the system to recover from all possible recoverable faults, and capturing and storing images via the USB payload interface. Future revisions will include updates to the PiRL logic to ensure preset startup states of the flight computer processors and boot SD card. The BFC is scheduled to fly on the Beavercube CubeSat in October 2020.

Acknowledgements

The authors would like to Prof. Kerri Cahoy of MIT AeroAstro and members of MIT's STAR Lab, especially Bobby Holden whose work on the DeMi CubeSat flight computer inspired the initial design of the BFC, MIT UROP participant Brandon John for support on schematic design and contributions to the final board layout, and Paul Fucile from the Woods Hole Oceanographic Institute for support on board design review and hardware testing.

References

[1] R. E. Morgan P. do Vale Pereira, B. G. Holden. Calibration and testing of the deformable mirror demonstration mission (demi) cubesat payload. In *Proceedings of the 33rd Small Satellite Conference*, 2019.

[2] V. Lappas C. P. Bridges J. D. Baker C. Underwood, S. Pellegrino. Using cubesat/microsatellite technology to demonstrate the autonomous assembly of a reconfigurable space telescope (aarest). *Acta Astronautica*, 114, 2015.

[3] Martha O'Bryan. *Radiation Effects and Anaylsis*. NASA, 2015.

Appendix A

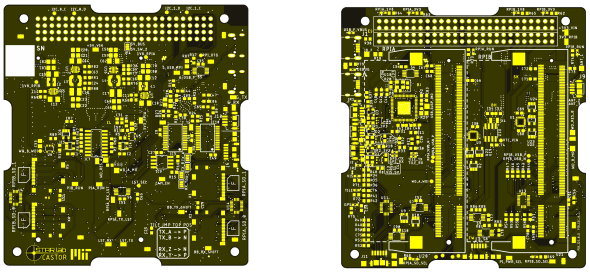


Figure 6: Beavercube Flight Computer Board Rev 1.0. Key features include 2 onboard Raspberry Pi (RP) compute modules, 4 switching voltage regulators for RP power, embedded micro-USB flash port, 7-port USB hub, 4 micro-SD card slots