Alex Trestyn (act88) 05/05/17
MAE 4900 Professor Mason Peck

CisLunar Explorers End of Semester Report – Spring 2017 – Power Subsystem

Introduction

This past spring was my first semester performing work for the CisLunar Explorers Electrolysis Propulsion Research Team. I primarily worked on the nanosatellite's Power Team Subsystem, conducting FlatSat operations along with Daniel Kim (dsk252), integrating many of the key components necessary for mission success. In addition, I have also worked with Alex Wong (aw528), reviewing Command and Data Handling (CDH) code used to monitor housekeeping data, power and propulsion controls, and data packet formatting. The Power Subsystem was overseen by Kyle Doyle (kpd43), the project leader of the CisLunar Explorers.

Throughout the semester, the Power Team Subsystem met every Monday to discuss the developments regarding FlatSat integration and testing, and after the meetings concluded, Alex Wong, Daniel Kim, and I stayed to review CDH code. Every Thursday, a full team meeting was held in the conference room on the fourth floor of Rhodes where each subsystem would update their counterparts on the status of their subsystems, and discuss matters regarding inter-subsystem cooperation, and project deadlines and updates. In addition, I had scheduled meetings on Fridays with Filipe Pereira (fmd43), to discuss the Communications (Comms) architecture of CisLunar.

(Note: Many of the documents linked to here are hosted in CisLunar's folder on Cornell Box. Due to revisionary and organizational purposes, documents are frequently updated, moved, or deleted. If any of the documents here cannot be located at the links provided, please contact Alex Trestyn at act88@cornell.edu for copies if the documents cannot be found).

FlatSat Testing

Instead of trying to test satellite hardware (mechanical and electric), in conjunction with satellite software while all components are part of a consolidated stack, it is far more efficient to test inter-connected hardware laid out flat, or "two-dimensionally," in FlatSat configuration. During this semester's work, Daniel and I were able to successfully incorporate all the necessary hardware components of the satellite. At the center of the FlatSat were the Raspberry Pi (RPi) Model A+ Flight Computer, and the GOMSpace EPS battery, the latter of which I was vaguely familiar with due to my time spent working on the OAAN research team. The datasheets used for testing and integrating these hardware components have been uploaded here: https://cornell.app.box.com/files/0/f/27210882420/Datasheets

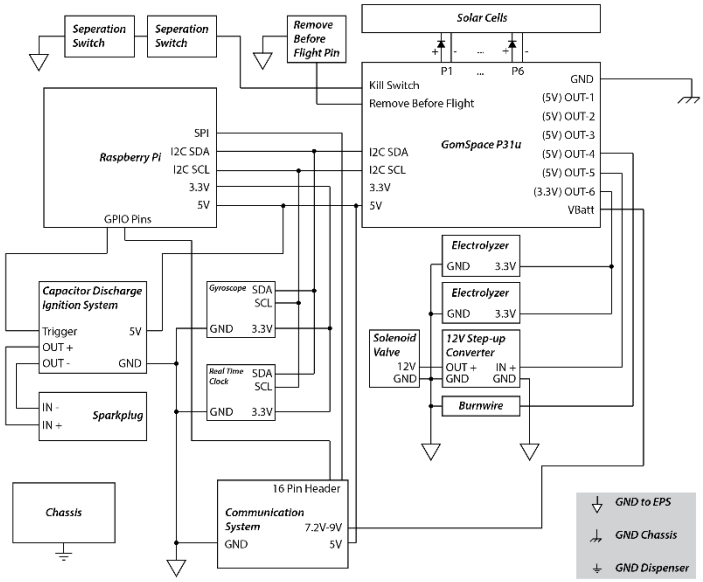

Figure 1 – Diagram of the FlatSat interconnect (credit to Daniel Kim)

The electrolyzers used in the experiment were Horizon FCSU-010 Mini PEM Fuel Cells. These were tested by filling a single beaker with distilled water, which in turn was placed inside a larger, stable container to prevent any damage to the satellite's electronic hardware in the event that water spilled from the container. Both electrolyzers were connected to OUT-6 (Pin H1-52) of the battery and fired simultaneously. Upon repeated testing, it was observed that multiple firings may be necessary as the electrolyzers take a bit of time to fully "warm up" and be able to properly take in water to electrolyze.

The solenoid valve was powered via an Adafruit ADS1115 ADC converter, which was connected to the battery output pin OUT-5 (H1-50). To test that the solenoid valve was working correctly, a brief voltage pulse was applied from the battery output through the ADC, and in turn the solenoid valve would release a brief puff of air. Video footage of this was recorded and upload to the following page:
https://cornell.app.box.com/files/0/f/20827152026/FlatSat_Testing_Videos

The spark plug was implemented by having it hooked up to the output of a Capacitor Discharge Ignition (CDI) system, which in turn was controlled by the RPi output. To ensure that the spark plug was properly integrated, a brief triggering signal was sent to the CDI, and a spark was observed on the sparkplug. The gyroscope/accelerometer (LSM303DLHC, from STMicroelectronics), was tested by re-orienting its position and moving (accelerating) the device, and observing the changes in transmitted data.

One of the more difficult components to integrate as part of the FlatSat was the nichrome burn wire. The specific nichrome wire used as part of the CisLunar FlatSat was TEMCo RW0282 30 AWG wire. In order for the wire to cut the restraining fishing wire to allow the two halves of CisLunar to separate, it needs to "burn" at a high enough power level. In the lab setting, the burn wire was measured to have a resistivity of approximately 0.22 Ω/cm. at room

temperature. After repeated testing, it was determined that in order to consistently cut through the fishing wire, the length of burn wire selected would need around 2.0 A of current running through it. Therefore, the allottable current range is between 2.0 A and 3.0 A, as the GOMSpace EPS battery will latch-up if any of its outputs reaches the 3.0 A mark.

To conduct FlatSat testing, it was necessary to SSH into the RPi Flight Computer to send commands via user console to examine individual components. Based on the pre-existing code written by Daniel Kim, I wrote up a guide to instruct new users on how to properly wirelessly SSH into the Raspberry Pi and how to set up GPIO pins to send output signals – link: https://cornell.app.box.com/files/0/f/27196342425/1/f_173108609596

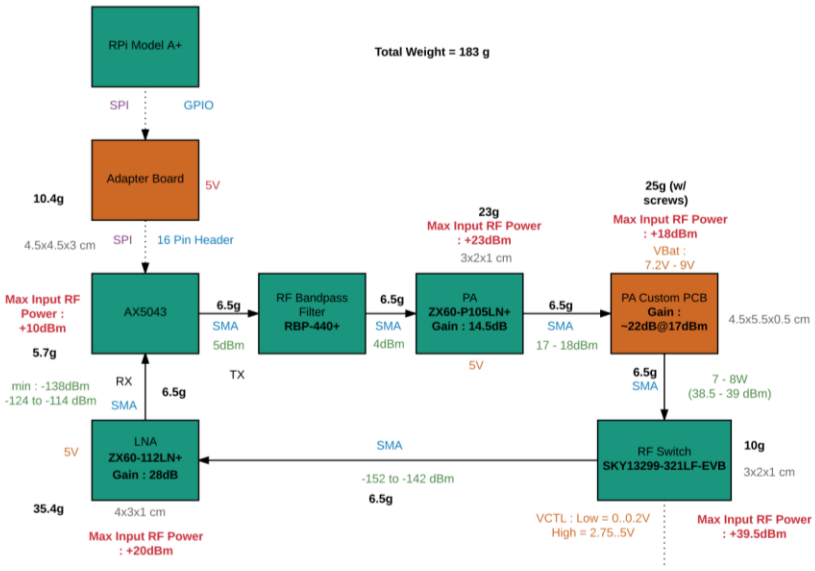

Figure 2 – Comms System Hardware Architecture (image created by Filipe Periera)

One of the last pieces of hardware integrated was the Comms System transmitter/receiver. Filipe Periera met with members of the Power System to help explain the interconnect between the individual components and how testing should be done for FlatSat purposes.

For the GT4 competition submission, a master folder including relevant documents, diagrams, and test procedures was compiled. Various images from the FlatSat integration phase were uploaded as part of this submission, and can be found here: https://cornell.app.box.com/files/0/f/20827153361/FlatSat_Testing_Images

GT4 Testing Submission

As part of CisLunar's participation in NASA's Cube Quest Challenge competition, I helped to create, edit, and procedurally conduct the necessary Test Verification Reports that would cover various aspects of nanosatellite operation that fall under the Power Subsystem, along with Daniel Kim. These tests revolved around satellite separation, the various conditions

surrounding charging, and power consumption. The specific tests conducted were outlined in the CisLunar VCRM Excel document created by Lindsey Hayes, which can be found here: https://cornell.app.box.com/files/0/f/17264589296/1/f_156195690379. The finished Verification Tests that we had prepared and performed (3-6, 20, 47, 48, 51, 52) are located here: https://cornell.app.box.com/files/0/f/23369122267/Test_Reports.

In addition to preparing Verification Reports, I worked on the master GT4 CubeQuest Design Document (https://www.sharelatex.com/project/58d45ed8207308275cf317fd). I was tasked in particular with editing and reviewing the "Communications Subsystem," "Electrical Power System," and the "Command and Data Handling/Flight Software" sections of the document (Sections 4.1-4.3, respectively).

The largest obstacle in completing GT4 testing was due to test procedures involving the vacuum chamber located in the graduate laboratory workspace in Rhodes 450. Although the vacuum chamber was able to provide an adequate depressurized testing space, there were significant problems involving tests that required an external voltage/current source. When trying to supply an external 3.7 V and 1.0 A to test battery charging, a current rated at less than 0.2A was measured at the receiving end on the inside of the vacuum chamber. Based on the connection this would imply that somehow, current was being lost to an unknown sink, even though this setup could essentially be modeled as a series circuit where input current equals output current. Numerous testing configuration were attempted, including different DC power supplies, different connection points, and even external batteries placed inside the vacuum chamber, most of which were unsuitable for testing, and wasted a significant amount of time during the GT4 submission period. I highly recommend that in advance of future vacuum chamber testing that utilizes external power sources, the wiring, connections, and circuitry of the vacuum chamber be replaced and tested to make sure there are no connectivity problems or current leakages. Prior to making any changes to the vacuum chamber circuitry, Ryan Elandt (rbe38), who was involved in the vacuum chamber setup, should be contacted for consulting regarding this issue. In regards to vacuum chamber operation, I compiled a written guide to detail the procedure and safety hazards involved with testing in the chamber – link: https://cornell.app.box.com/files/0/f/27196342425/1/f_173003154761

CDH CSC Reviews

The file "ccsds_header.py" is responsible for creating the data packet format used in transmitting data to the Ground Station. The data packet format was adapted from the CCSDS packet format, but cut down to limit packet size over bandwidth concerns. "hs_manager.py" is used to grab relevant housekeeping data from the different nanosatellite systems and components (e.g. power subsystem, propulsion subsystem, camera functionality, flight computer status, etc.). "propulsion_controller.py" largely deals with the Propulsion Subsystem, including cold gas thrusts, electrolyzer control, the spark plug, and burn wire activation.

To aid Alex Wong with his software development, Daniel and I estimated the approximate current and voltage consumption of each satellite hardware piece. This was done by comparing the housekeeping data obtained with each piece connected to the EPS battery to the

housekeeping data obtained after each hardware component was disconnected. The results were tabulated for each component and can be found in the following link:
https://cornell.app.box.com/files/0/f/649402197/1/f_173058650633

Earlier in the semester, I worked with Filipe Periera and Alex Wong regarding the software for the AX5043 Transmitter. Filipe originally wrote code for the transmitter in C++, and Alex later translated the code to Python. I helped Filipe compare the two files, and fix any errors in translation, as well as commenting/documenting the code for ease of use.

Other Work

Due to my previous work with the Violet Nanosatellite Project Team, I was able to aid with the CisLunar Verification Test Assignments for GT4 by procuring documentation from Violet's most recent reports. I was able to obtain End of Semester Reports from the CDH system, test procedure templates, guides, and other ITAR documentation that will be of help to CisLunar. In addition, I uploaded a copy of the core Flight Executive software guide that was given to Violet by NASA. The aforementioned documentation can be found in the "Violet Reference Documents" folder on Cornell Box, linked to below:
https://cornell.app.box.com/files/0/f/24102943822/Violet_Reference_Documents
Thanks to Kevin Hui (kh523) for his assistance in procuring documentation and answering questions regarding this matter.

As a habit from previous semesters spent working on the Violet Project Team and OAAN Research Team, I compiled a "CisLunar Acronym List" document, to help keep track of any new or significant information that I came across as I integrated myself into the team. Such reference documents can be helpful when trying to more efficiently and quickly integrate new members into the research team. Link:
https://cornell.app.box.com/files/0/f/27196342425/1/f_173112476566

In early March, I volunteered to help Kyle in his setup of the ground station in the upper levels of Rhodes. While Kyle was on the roof of Rhodes, I, along with another member of the Cornell facilities faculty, helped snake through the main co-axial cable from the roof into the top floor of the building, directly below.

Future Work

The next logical step following FlatSat integration would be trying to integrate the hardware components in stack formation – in the actual configuration in which they would be placed inside the satellite. This cannot occur until the appropriate machine satellite frame, or suitable 3D-printed model has been provided to the Power Subsystem for operation.

Several of the Verification Test Procedures for GT4 involved charging the GOMSpace battery via the solar-panels. However, were no adequate solar panel hardware pieces prepared for the Power subsystem at the time of testing. In place of the solar panels, DC power supplies at 3.7

V and 1.0 A were used to charge the battery through the photo-voltaic (PV) ports. After the solar panels have been obtained and cut to proportion for nanosatellite use, it is recommended that the relevant Verification Test Procedures be re-evaluated.

At the last one or two team group meetings, members of the Propulsion Subsystem have brought up concerns regarding the spark plug. It seems after multiple iterations of spark plug tests, team members are concerned about water condensation on the spark plug surface, citing that there was a build-up of water in the spark plug well. This may significantly impair the ability of the spark plug to properly fire. It has been proposed that a glow plug be implemented in place of the spark plug to significantly decrease the chance of a misfire.

For future testing involving the burn wire, it may be beneficial to incorporate a mechanical spring-loaded release mechanism to cut the fishing wire. In previous tests, the burn wire's cutting efficiency was tested more crudely by manually attempting to apply a constant force between the burn wire and the fishing wire.

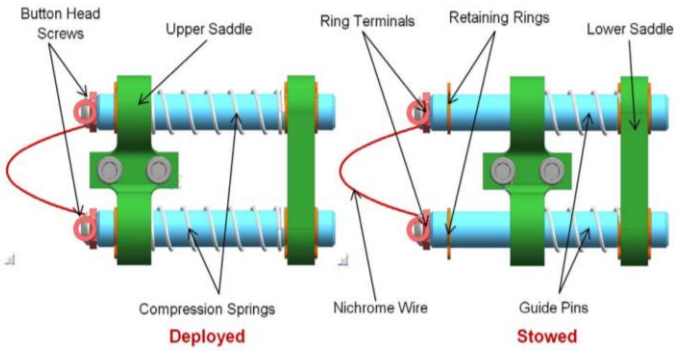

Figure 3 - Nichrome burn wire release mechanism

The figure above was incorporated from the article "A Nichrome Burn Wire Release Mechanism For CubeSats" by Adam Thurn, et al. Implementing such a release mechanism would eliminate some uncertainties with future burn wire testing, which is crucial for mission success. The paper referenced can be found at the following link: http://bit.ly/2quJbCg