OpenOrbiter: a Student Space Program

Jeremy Straub
Department of Computer Science
University of North Dakota
Grand Forks, ND 58202
jeremy.straub@my.und.edu

Joshua Berk
Department of Space Studies
University of North Dakota
Grand Forks, ND 58202
joshua.berk@my.und.edu

Christoffer Korvald
Department of Computer Science
University of North Dakota
Grand Forks, ND 58202
christoffer.korvald@my.und.edu

Abstract

OpenOrbiter is a student-conceived, student-run small spacecraft program operating at the University of North Dakota. It involves students from numerous departments including both STEM (computer science, electrical engineering, mechanical engineering, space studies) and non-STEM (business, entrepreneurship, education) disciplines. The program is comprised of 19 student-lead, faculty-mentored groups focusing on all areas of spacecraft design and fabrication.

This paper presents the OpenOrbiter space program as a model for emulation. It focuses specifically on the utility of participation for students and faculty in the Computer Science discipline, while briefly summarizing the motivation for participants in other disciplines. These benefits include learning the vernacular of other disciplines that students will need to work with in careers in industry, being part of a team project with demonstrable results and gaining a valuable resume item that is inspirational to prospective employers. Participants also learn how to work in an industry-relevant environment, skills related to working on a large project and how to collaborate with team-members that can’t have a group meeting over every single decision.

Work to-date has included the completion of mechanical designs, high-level electrical designs and software designs. Work on software implementation is currently ongoing. This work has been presented in numerous papers at numerous conferences and in several journal articles.

While significant work has already been done, much more remains ahead. The paper, thus, concludes with a focus on future work. This includes the fabrication of hardware components, software development, unit and integration testing and launch provider compliance activities. A significant focus on documentation also exists, as these documents will become the core of the Open Prototype for Educational NanoSats design document kit.

1. Introduction

A CubeSat is one of the smallest sizes of spacecraft; a 1U (one-unit) CubeSat has the dimensions 10 cm x 10 cm x 10 cm and can weigh no more than 1.33 kg. Other CubeSat sizes follow this unit standard. Common sizes include 2U (10 cm x 10 cm x 20 cm, 2.66 kg), 3U (10 cm x 10 cm x 30 cm, 4 kg) and 6U (10 cm x 20 cm x 30 cm, 8 kg). Despite its small size, a CubeSat performs all of the major functions undertaken by larger spacecraft (except propulsion, which is generally present in most spacecraft and currently not allowed under NASA integration standards). Because of this, CubeSats serve as an excellent training platform for the engineers, managers and software designers and developers for future larger spacecraft. They were originally developed by Jordi Puig Suari and Robert Twiggs in the late 1990's [1]. Now, due to miniaturization of electronics, they are also a target platform, in their own right, for science, military and other government applications.

The Open Prototype for Educational NanoSats (OPEN) [2] seeks to make these small spacecraft, which generally cost $50,000 to $250,000 to build [3] even more affordable. OPEN is developing a set of design documents, assembly instructions and software which will be made available to anyone worldwide that wishes to build a CubeSat-class spacecraft. The OPEN program is poised to allow the creation of a CubeSat-class spacecraft (excluding mission-specific payload elements) with a parts budget of under $5,000 [2, 4]. By doing this, OPEN makes CubeSats affordable to institutional teaching funds, small faculty projects (e.g., seed and startup funding).

The OpenOrbiter spacecraft will be the test mission which will demonstrate the space worthiness of the OPEN designs. This is a critical step to building acceptance for them in the small spacecraft community. In addition, it will demonstrate the University of North Dakota's (UND) capabilities as a leader in this area.

Work to-date has included the completion of mechanical designs, high-level electrical designs and software designs. Work on software implementation is currently ongoing. This work has been presented in numerous papers at numerous conferences and in several journal articles. This process has provided 20 students working in the software groups and over 200 students involved in the project with significant experience in a variety of technical and non-technical disciplines.

2. Background

The creation of a spacecraft is, inherently, an interdisciplinary process. The demands of the space environment and spacecraft insertion into this environment require a higher level of integration than with many other types of interdisciplinary projects. Spacecraft, generally, face limitations on their mass and volume as well as demanding power requirements forcing tradeoffs between the operations of various subsystems to keep the spacecraft utilizing power within its solar generation capabilities [5]. The spacecraft also faces significant thermal constraints: it must remain within a narrow margin of generating too much heat and overheating components and generating too little and freezing in the cold vacuum of space [5]. The correct solutions for subsystems are not known a priori; they must be determined through an iterative process [see 5-7] that propagates the impact of a change in one area to all other affected areas and verifies that each change has not violated requirements or constraints in these areas (corrective action must be taken, if a violation occurs).

Students, who are generally being silo-trained (i.e., not exposed to projects that involve collaboration outside their major discipline) are not well suited to enter the workplace in the absence of additional training in interdisciplinary working techniques. They must also learn the vernacular of the various other disciplines that they must collaborate with, in order to be able to communicate effectively with practitioners from these disciplines. Unfortunately, in many cases, this aspect of the educational experience is neglected at colleges and universities – being left, instead, to employers to perform.

Student learning has been shown to be aided by participation in inspirational, hand-on projects. These projects allow students to reinforce their classroom learning via utilizing the knowledge and skills in a closer-to-real-world environment. Students also have the opportunity to learn new skills: technical, managerial and interpersonal.

3. Work To-Date

Work on the OpenOrbiter project commenced with the decision as to what area to focus on (the goal of creating a CubeSat pre-existed initiation; however, initial discussions and evaluation focused on what the goal of the program should be: a specific science objective, engineering objective, etc.). With the focus (creating an enabling platform for others) determined, a mission plan was created. Concurrently with this, a mission name (OpenOrbiter) and branding were developed.

With an initial mission plan created, work turned to delivering on this plan. Initial work focused on progressively lower-level design. Starting with high-level requirements, each team (e.g., the eleven electrical, mechanical and software groups) created their own group-level goals and objectives that would fulfill these requirements. From this, they created hardware or software designs that would be utilized to guide implementation.

A. Electrical & Mechanical

The electrical teams are responsible for the development of numerous critical systems for the spacecraft. Electrical teams include power generation and distribution, communications, attitude determination and control, optical payload development, and sensor and bus development. These teams have created initial design documents and now have progressed to board-level design. When the board designs are done, they will be fabricated by a PCB fabrication vendor and components will be mounted on them (in appropriate locations) by students. The finished boards will be tested on the vibration table and in the thermal vacuum unit to verify their suitability for use in space and compliance with CubeSat integration guidelines.

The mechanical team has designed the overall structure of the spacecraft. They have created CAD and other models of this design. Current work relates to 3D printing this design for additional verification and validation of its suitability (and to provide a model for the electrical teams to work with).

B. Operating Software

The Operating Software team has developed a complete high-level and low-level design of the operating software. They have implemented code to communicate with temperature sensors, GPS and the spacecraft's camera. They are currently working on the development of the main control software routine which will manage the minute-to-minute operations of the spacecraft. The Operating Software team has also collaborated with the ground station team to develop communications and interoperability standards for task assignment and data transmission to and from the ground. They have also coordinated with the payload processing team to coordinate data formats and message passing between the two onboard software systems (which run on different hardware).

C. Payload Software

The Payload Software team has completed a set of TCP/IP routines that will be utilized for onboard message passing between the payload and operating software (between their two different computing platforms). They have also performed research and analysis related to super resolution and mosaicking. They have compared the SIFT and SURF algorithms for feature identification and image alignment, with a particular emphasis of determining how a priori information (e.g., from the GPS and inertial measurement unit) can be utilized to make the mosaicking process more efficient. In addition, they have performed work on maximizing the capabilities of the GumStix WaterStorm COM units that the payload software will run on. They have also researched and developed code for low-computational-cost orbit projection.

D. Ground Station Software

The Ground Station software team has developed a data storage mechanism for storing commands awaiting transmission to the spacecraft and data returned from the spacecraft, which will need to be disseminated to users. They have developed a user interface and web-based front end code for the user-facing command console. They have also developed PHP code to connect this web-based interface to back-end code, written in C++, which will perform the actual control of the ground station hardware to point it at the spacecraft and send and receive messages to/from the spacecraft.

E. Testing and Validation

The initial work of the Testing and Validation team (which was initiated later than the other teams) has focused on developing tester familiarity with system operations. From this and design documents, the team is developing a test plan document.

F. OpenEdge

The OpenEdge team is working to develop an expert system to assist OPEN users in configurating their spacecraft. Once this software is done, board developers, users and others can load in the details of their hardware. This will allow constraint violations, space incompatibilities and other issues to be checked for prior to detailed design and development. This should reduce cost via the elimination of redundant work (or work for which the value will not be recognized due to it being unable to be included in a final product). OpenEdge will provide feedback on whether a design works or not and will enumerate the specific rules or conditions violated, if a design is determined to be non-compliant, to facilitate correction.

4. Relevance to the Computer Science Discipline

The question of why building a spacecraft is relevant to computer science is important to understand and consider. This answer is multi-faceted. First, students of computer science will (generally) be graduating from the abstracted world of the university and starting work in a real-world environment where they will be required to interact with things (i.e., hardware, users) that do not always work the way they should or the way they have been simulated. To this end, the small spacecraft program provides them with a low-risk opportunity to familiarize themselves with this dilemma before workforce entry.

Second, employers (generally) seek staff members who are able to communicate their ideas, needs and suggestions effectively. Participation drives these development aspects for students.

Third, the hands-on, real-world project with a very emotive outcome (the launch of the craft into space) is inspirational to students. This drives their interest in the computer science field (and other STEM fields).

Finally, the spacecraft is inherently a platform for software operations. This viewpoint has not always been accepted (particularly in the days where extraordinarily low capabilities required the software to do little more than enact controller instructions). However, the miniaturization of electronics (including processors, storage, sensors and actuators) means that system design can be driven by software requirements (and the software requirements implement the system-level requirements provided by the various stakeholders).

5. Qualitative Evaluation and Anecdotal Evidence of Success

Students who have been working on the project have evidenced a number of positive outcomes [8]. First they have demonstrated a greater ability to perceive the system as a whole (i.e., an integrated collection of interdependent and highly-related parts). Many have gained and demonstrated proficiency in a new programming language (e.g., C++, PHP). All of the students have had to learn how to utilize the Git version control software and the GitHub management system.

Involved students have also demonstrated a greater understanding of the importance of testing code and keeping the testing process (and through this, the requirements and constraints) in mind while developing. For many students, this was their first experience working on an open-source project (and thus requiring them to write ‘clean’ easily-understandable code that would be reviewed and used by others). Students have also, to this end, had to learn how to write documentation that was useful to not just themselves and their teammates, but to others (within or external to the project) whom might utilize the code and documentation in the future.

Students have also evidenced a greater level of comfort with and proficiency at communication. Team members that were shy at the beginning of the project (and who may not have contributed much in class) now communicate regularly with their team members and those from other teams project-wide. Many students (including a number of undergraduates) have also made (or will soon make) project-relevant presentations at national and international conferences. To this end, they have developed speaking and presentation skills. Perhaps most importantly (with regards to communications), team members are learning (and evidencing their ability to) communicate effectively with regards to technical subjects with those from other disciplines.

6. Future Work

Future work focuses on the completion of the OPEN software, designs and documentation and the completion of the OpenObiter spacecraft. The documentation will include written information as well as online videos which must be scripted, recorded and post-processed. Designs will include CAD and PCB diagrams and software design documents. Software will include full source code for the operating and ground station software developed for the OpenOrbiter project.

The OpenOrbiter spacecraft is targeted for completion towards the end of 2013 or early 2014. It is hoped that the spacecraft could be launched into orbit via the NASA Educational Launch of NanoSats Program (ELaNa) during 2014. Once the spacecraft is launched, efforts will focus on operational control of the spacecraft and assessment of its performance.

In the short term, future work also relates to the assessment of team member learning and benefit from participation in the project. To this end, assessment activities are currently being designed and will be administered to team members.

7. Conclusion

This paper has presented an overview of the work performed on the OPEN designs and OpenOrbiter spacecraft to-date. It has highlighted the benefits of program participation for the numerous students involved (both from the Computer Science discipline and others). Limited qualitative and anecdotal evidence has been presented to assist reader understanding of the benefits attained by the participants. Future work will quantify and refine this limited assessment.

References

[1] R. Twiggs and B. Malphrus. 2011. CubeSats. In Space Mission Engineering: The New SMAD. J. Wertz, J. Puschell & D. Everett, eds., pp. 803-821.

[2] Berk, J., J. Straub and D. Whalen. 2013. Open Prototype for Educational NanoSats: Fixing the Other Side of the Small Satellite Cost Equation. Proceedings of the 2013 IEEE Aerospace Conference.
[3] Straub, J. 2012. Cubesats: A Low-Cost, Very High-Return Space Technology. The proceedings of the 2012 Reinventing Space Conference.
[4] Straub, J., C. Korvald, A. Nervold, A. Mohammad, N. Root, N. Long, D. Torgerson. 2012. OpenOrbiter: A Low-Cost, Educational Prototype CubeSat Mission Architecture. Machines, Vol. 1, No. 1.
[5] Wertz, J., J. Puschell & D. Everett, eds. 2012. Space Mission Engineering: The New SMAD. El Segundo, CA: Microcosm Press.
[6] Wertz, J. & W.J. Larson, ets. 1999. Space Mission Analysis and Design, 3rd Edition. El Segundo, CA: Microcosm Press.
[7] Fortescue, P., G. Swinerd & J. Stark. 2011. Spacecraft Systems Engineering. New York, NY: Wiley.
[8] Straub, J., J. Berk, A. Nervold and D. Whalen. 2013. OpenOrbiter: An Interdisciplinary, Student Run Space Program. Advances in Education, Vol. 2, No. 1.

Acknowledgements

Small satellite development work at the University of North Dakota is or has been supported by the North Dakota Space Grant Consortium, North Dakota NASA EPSCoR, the University of North Dakota Faculty Research Seed Money Committee, North Dakota EPSCoR (NSF Grant # EPS-814442), the Department of Computer Science, the John D. Odegard School of Aerospace Sciences and the National Aeronautics and Space Administration. The participation of numerous students and faculty spanning multiple UND departments in the small satellite program is gratefully acknowledged. Corporate sponsors, including GitHub Inc. who currently provides their GitHub service to the OpenOrbiter project at no charge, are also greatly appreciated. RedMine is provided to the OpenOrbiter project by Trollweb A/S (Norway) who installed and host this software at no charge to the project.