CubeSat mission design based on a systems engineering approach

Conference Paper *in* IEEE Aerospace Conference Proceedings · March 2013

DOI: 10.1109/AERO.2013.6496900

2 authors:

Sharan Asundi
Old Dominion University

Norman Fitz-Coy
University of Florida

CubeSat Mission Design Based on a Systems Engineering Approach

Sharan A. Asundi
Aerospace Science Engineering
Tuskegee University, Tuskegee, AL, 36088
334-727-8768
asundi@mytu.tuskegee.edu

Norman G. Fitz-Coy
Mechanical and Aerospace Engineering
University of Florida, Gainesville, FL 32611
352-392-1029
nfc@ufl.edu

Abstract— **With the exception of the CubeSat specification, CubeSat design and development approaches have been mostly ad hoc, which has questioned their reliability. A systems engineering approach, based on the guidelines of NASA's Systems Engineering Handbook has been developed for CubeSats to facilitate systematic design, development and address their reliability, traceability, and reusability. The CubeSat systems engineering approach, developed as a repeatable process, uses a top-down design methodology to translate mission definitions into basic building blocks, components, interfaces and tasks, that then facilitate a bottom-up development and fabrication process. Some of the design tools (e.g., N2 diagram) described in NASAs Systems Engineering Handbook are utilized early in the design phase to identify potential conflicts in the mechanical and electrical interfaces. A novel subsystem level flowdown, which transcribes the system level requirements into identifiable CubeSat subsystems, (i.e., building blocks) is described. Utilizing this approach yields full traceability from mission concept to subsystem component to flight software. Additionally, the approach facilitates the estimation of the mission overhead in terms of power, telemetry, and computation associated with each component, interface, and task.**

TABLE OF CONTENTS

1. INTRODUCTION

CubeSats [1], [2], conceived as educational satellites, are challenging the paradigm of traditional satellites and are being recognized for their potential utility [3], [4], [5], [6], by space and research agencies around the world including United States (U.S.) National Aeronautics and Space Administration (NASA) [7], [8], [9], U.S. Department of Defense (DoD) [10], National Science Foundation [11], [12], and the U.S. National Reconnaissance Office [10]. Orbital debris disposal has been identified as a serious concern and guidelines have been produced by NASA and other space agencies for enforcing it for satellite missions [13]. However, much of the research and development of established systems for de-orbiting satellites, is focused on larger spacecraft, particularly in the geosynchronous Earth orbit. The experimental purpose of CubeSat missions, lack of de-orbiting systems, low radiation levels which has enabled the use of high grade commercial-off-the-shelf (COTS) components for design, development, has resulted in targeting these class of satellites for low Earth orbit missions. With the use of high grade COTS based technologies, reduced development cost, and time, CubeSats have made space more accessible and space missions highly desirable. However, their design and development approach has lacked a systems engineering basis and questioned their reliability. To address this issue, a systems engineering approach is presented for the design and development of CubeSats. The approach, based on the guidelines of (NASA's) Systems Engineering Handbook [14], is described in the context of SwampSat [15], [16], a University of Florida 1U CubeSat, to limit the abstract discussion.

A top-down design approach is developed as a repeatable process for translating the mission objectives and mission requirements into hardware components, interfaces and operating tasks implementable as software routines. The top-down design approach facilitates a bottom-up development and fabrication process and addresses the reliability of these class of CubeSats through a traceability map. In addition, the overhead in terms of power, telemetry and computation associated with each component, interface, and task facilitates the estimation of the mission overhead for these factors. It is important to note that the mission overhead does not address any financial costs associated with these class of CubeSats.

SwampSat is a 1U CubeSat technology demonstrator with the mission objective of demonstrating rapid retargeting and precision pointing (R2P2) [15] capability for pico- and nanosatellites. The mission objective specifically serves as an example of how 1U CubeSats can support Tier-3 objectives of U.S. Department of Defense's (DoD's) Operationally Responsive Space (ORS) program [17]. A successful mission will provide flight heritage to the mission payload, advance its technology readiness level and in the process address the challenges of DoD's ORS program, and NASA's Franklin and Edison programs. The attitude control system (ACS) is designed around small scale control moment gyroscopes (CMGs) and the mission objective encompasses precision pointing for remote observation [18]. Specifically, the mission objective will demonstrate precision three-axis attitude control with an experimental ACS consisting of four single gimbal CMGs [15], [19], [20].

978-1-4577-0557-1/12/$26.00
[1] IEEEAC Paper #2180, Version 2, Updated 05/11/2012.

2. CUBESAT SYSTEMS DESIGN AND DEVELOPMENT

The block diagrams shown in Fig. 1 and Fig. 2 provide an overview of the approach. As shown in Fig. 1, mission definition is the starting point of the design process. A well-articulated mission definition leads in to identification of specific primary and secondary mission objectives. Although, mission objectives are the primary influence for defining requirements and designing concept of operations (CONOPS), the external drivers, which include financial cost, schedule, constraints, and lessons learned from previous missions or parallel systems design and development, could have a significant influence. Specific mission requirements, both functional and performance, which are described in technical terminology, are identified and associated with the mission objectives. The mission CONOPS is designed to describe the procedure for validating mission objectives on orbit.

The subsystems block, expanded in Fig. 2 to show the categories of each subsystem, can be thought of as a coin sorter machine. The system requirements when flowed through this block get broken down into components, interfaces, and tasks as basic building blocks. Each component is associated with one or more interfaces and the tasks, which involve one or more components and their interfaces, are grouped together to form operating modes as per the mission CONOPS. To establish a basis for calculating the mission overhead, each component is associated with a telemetry, and power overhead. Additionally, each task is associated with a computational overhead, which can be defined using an absolute scale (e.g., FLOPS) or a scale specific to the satellite computing platform (e.g., MSP430, C8051F120, TI DSP, ARM, PIC24). It is important to note that an operating mode will perform one or more tasks and each task will involve one or more components and their interfaces. One or more tasks may be common across operating modes and one or more components (and interfaces) may be common across tasks. The various levels of the requirements flowdown in Fig. 1 are described in the following subsections.

3. MISSION DEFINITION, OBJECTIVES, AND REQUIREMENTS

The mission definition, which is the starting point of the mission mapping process (Fig. 3), is captured as R2P2 terminology. Mission definition and mission objectives are described using non-technical terminology. High-level requirements are "allocated" to accomplish objectives and to facilitate detail design, requirements are "derived" where needed. The allocated requirements directly flow down from the mission objectives and may be too general to be directly implemented in the system design. The allocated requirements are detailed out to form the derived requirements, which could include

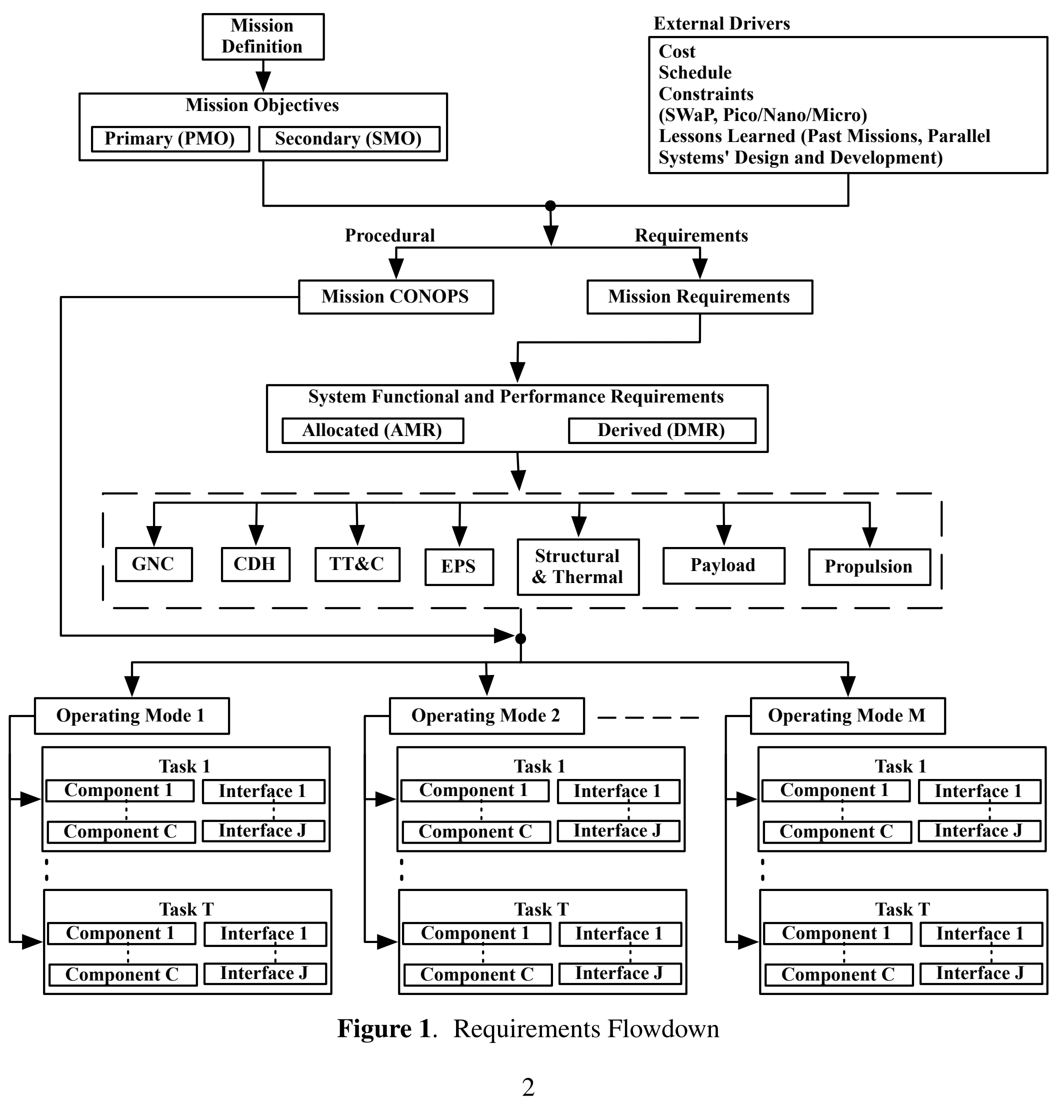


Figure 1. Requirements Flowdown

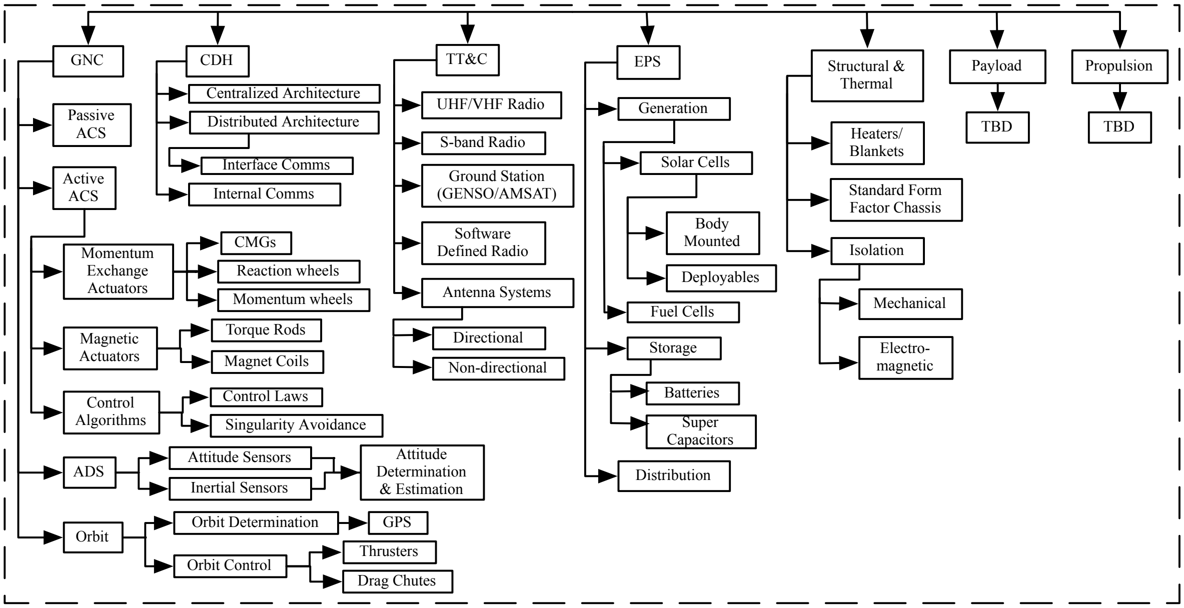


Figure 2. Subsystem Level Flowdown

new requirements for realizing the allocated requirements. To describe the mapping process, the primary mission objectives are labeled from PMO-A1 through PMO-A4, the secondary mission objectives from SMO-B1 through SMO-B3, the allocated mission requirements from AMR-A1 through AMR-A13 and the derived mission requirements from DMR-B1 through DMR-B19. The mission objectives and mission requirements are explained/described in detail in 21.

4. Mission Mapping - Mission Definition to Components, Interfaces, and Tasks

As shown in the requirements flowdown in Fig. 1, the mission definition is mapped to mission objectives, which are then mapped to mission requirements followed by other mappings, which ultimately translate to components, interfaces, and tasks. The mapping process (Fig. 3) is illustrated for one of the primary mission objectives (PMO-A2), which is derived from the mission definition and translated into basic building blocks. These basic building blocks include sensors, peripheral components, communication interfaces and implementable tasks. The mapping process ensures every building block, which can affect the mission overhead in terms of power, telemetry and computation, is derived from the mission definition and contributes towards achieving the mission goal. The complete set of primary and secondary mission objectives of SwampSat mission and the corresponding mapping is discussed in Reference 21.

5. Components, Interfaces and Tasks

The next step in the design process is the development of the N-squared (N2 or N^2) diagram, adopted from NASA's Systems Engineering Handbook [14]. The N2 diagram is used to capture the interfaces, mechanical and electrical, for all components of the satellite obtained through the mapping process. The electrical interfaces here include power and communication interfaces. The N2 diagram primarily

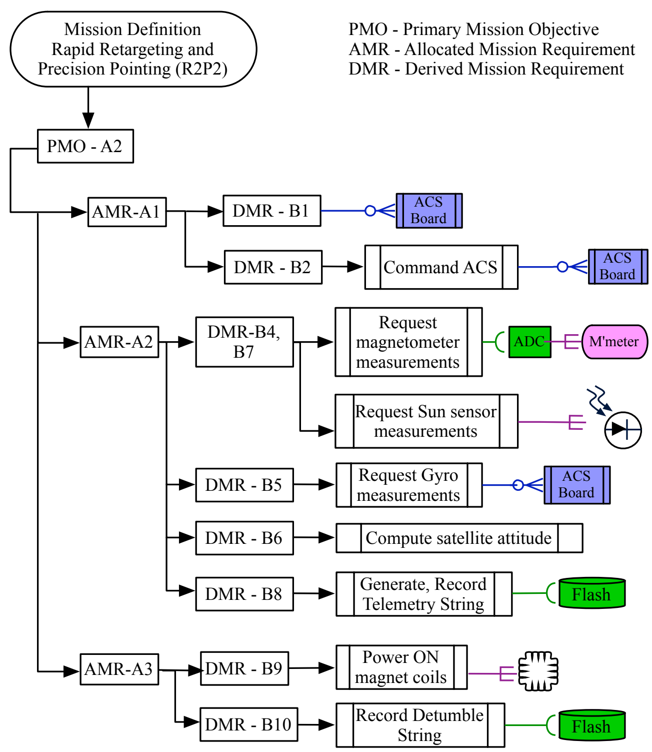


Figure 3. Mission Mapping Process

identifies areas where conflicts could arise in interfaces, and highlights input and output dependency assumptions and requirements [14]. A high-level N2 diagram, capturing the mechanical and electrical interfaces, is shown in Fig. 4. To illustrate the functionality of the N2 diagram better, consider the interface connections between the flight computer SFC430, magnetometer and Sun sensors. The triaxial magnetometer and six Sun sensors, together require nine analog channels to interface with SFC430. Since the SFC430 is designed around TI's MSP430 microprocessor, it can only accommodate six analog channels. The N2 diagram in this instance will

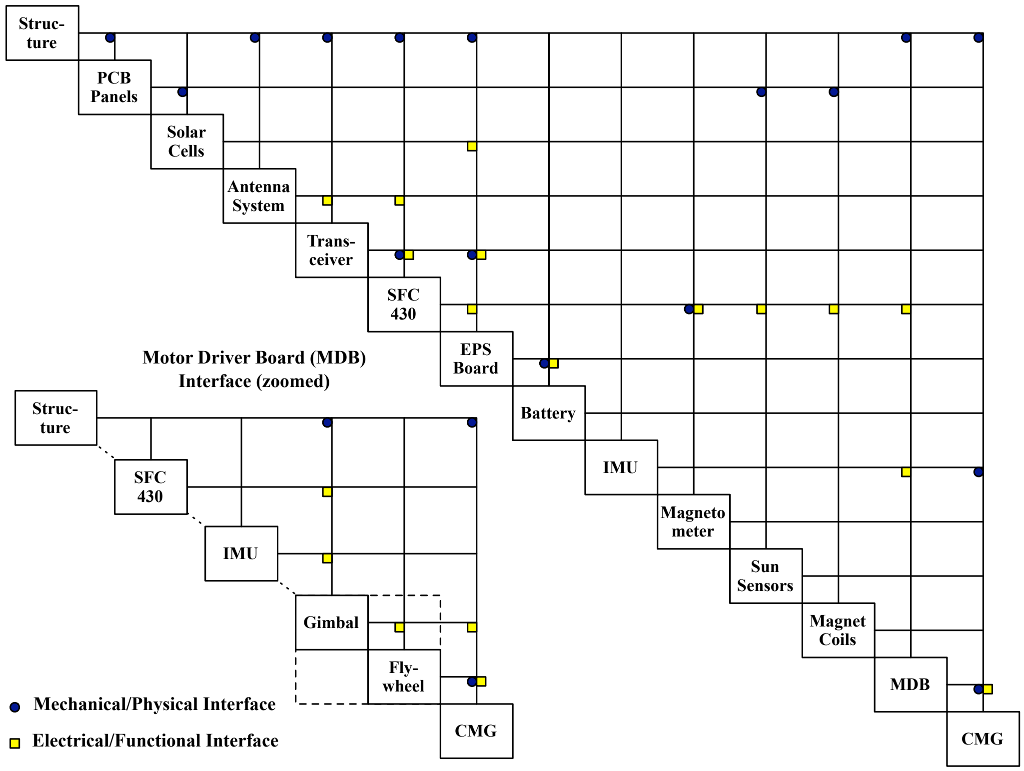


Figure 4. N2 Diagram

identify this limitation early in the design process and alert the system designer to seek solutions. It is important to note that interfaces can support one-to-one, one-to-many or many-to-many type connections. To ensure all components are accommodated, mechanically and electrically, on a satellite bus, a more detailed level N2 diagram could be used. The process of identifying specific communication interfaces through detailed N2 diagrams also facilitates protocol identification and implementation. The process of identifying non-conflicting components and their corresponding interfaces through the N2 diagram leads into developing a mission overhead model in terms of power, telemetry, and computation. To facilitate detail design, interfaces and components are associated with symbols, connector types, and color-coded. The communication interfaces identified through the mission mapping process demonstrated in Fig. 3 are shown in Fig. 5. As a special case, the communication interface between the satellite and ground station is included.

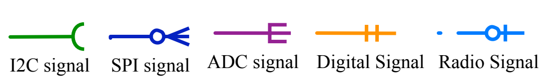


Figure 5. Interfaces

The components, which include subsystem boards, sensors, and peripheral devices, identified through the mission mapping process (Fig. 3) are shown in Fig. 6. Each component, sensor and peripheral device is associated with a symbol, an interface and a telemetry overhead where applicable. The tasks (Fig. 7) identified through the mapping process (Fig. 3) are linked with zero or more components and their corresponding interfaces. A computation overhead can be associated with each task to facilitate estimation of mission overhead discussed earlier. The process of assimilating components, interfaces, tasks and their associated overhead leads

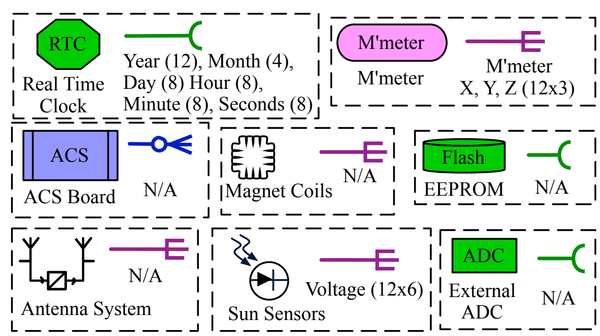


Figure 6. Peripheral Components, Sensors and Subsystems

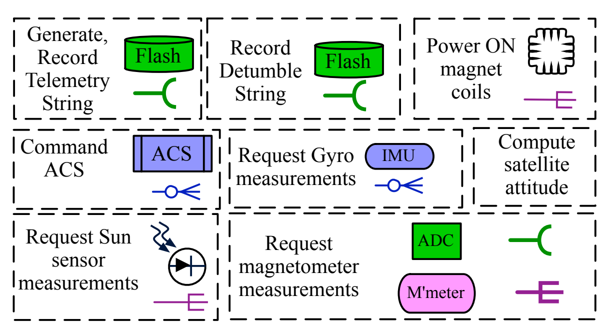


Figure 7. Tasks Implementable as Software Sub-routines

into their detail design in the form of subsystem designs. However, a critical element required for addressing the detail design is the mission CONOPS. A list of all the components, interfaces and tasks identified through the mission mapping process for SwampSat mission is presented in Reference 21.

6. MISSION CONCEPT OF OPERATIONS (CONOPS)

Mission CONOPS facilitates a systematic approach of realizing the mission objective and is critical for organizing mission operations. A mission CONOPS specialized for SwampSat mission, articulated in the form of a flowchart, is shown in Fig. 8. In essence, it captures the life cycle of a CubeSat and can be adopted as a layout for the design of flight software. The mission CONOPS also serves as an outline for identifying the phases or operating modes. These phases can be designed to be a one-time operation or operations, which can be executed multiple times. The mission CONOPS, illustrated in Fig. 8, is briefly summarized in the next paragraph.

The life cycle of the CubeSat is divided into 6 phases or operating modes. The power-up and deploy operation addresses a CubeSat's ability, as secondary payload, of not interfering with the primary payload during the initial minutes of launch [22]. The power-up and deploy mode also addresses the one time operation of antenna deployment and verification. The safe-hold mode, which is the CubeSat's primary operating mode, communicates real time telemetry to a ground station. The CubeSat can be operated in four other modes from within the primary operating mode. The detumble mode, apart from stabilizing the satellite, is designed to evaluate the performance of solar cells by recording their voltage, current and temperature. The ADS mode is designed to validate the attitude determination system and assist in diagnosing the attitude sensors on board the CubeSat. The CMG Ops mode validates the primary mission goal of rapid retargeting and precision pointing. To facilitate multiple attempts for realizing the mission goal, the operating modes can be executed multiple times.

The mission mapping process translates mission definition to components, interfaces and tasks. To illustrate the detail design phase of the CubeSat systems engineering approach, the building blocks obtained from the mission mapping process can be adopted to design the command and data handling system. Similarly, the telemetry overhead associated with the components can be utilized to formulate a detail telemetry budget. The detail design of one of the operating modes identified above and its associated telemetry budget is discussed in the next section.

7. DETAIL DESIGN OF OPERATING MODES AND TELEMETRY BUDGET

Driven by the mission CONOPS, the tasks derived from the mission mapping process are grouped together to form operating modes, which can be implemented as software routines. The safe-hold mode, which is one such operating mode, is designed to communicate real time telemetry to a ground station. The real time telemetry, which includes solar cell voltages, currents, temperatures, battery capacity, and satellite angular rates, relays satellite health and assists ground control in decision making. The downlink telemetry and the uplink command associated with the safe-hold mode are formulated based on the telemetry overhead associated with the components.

Safe-Hold Operating Mode

A flowchart illustrating the power-up & deploy mode and the safe-hold mode is shown in Fig. 9. The P-POD design [23] and the CubeSat requirement of a separation switch [24] enable the satellite to be powered on as soon as it is launched from the P-POD. The flight software is loaded into the computer memory and the satellite enters the power-

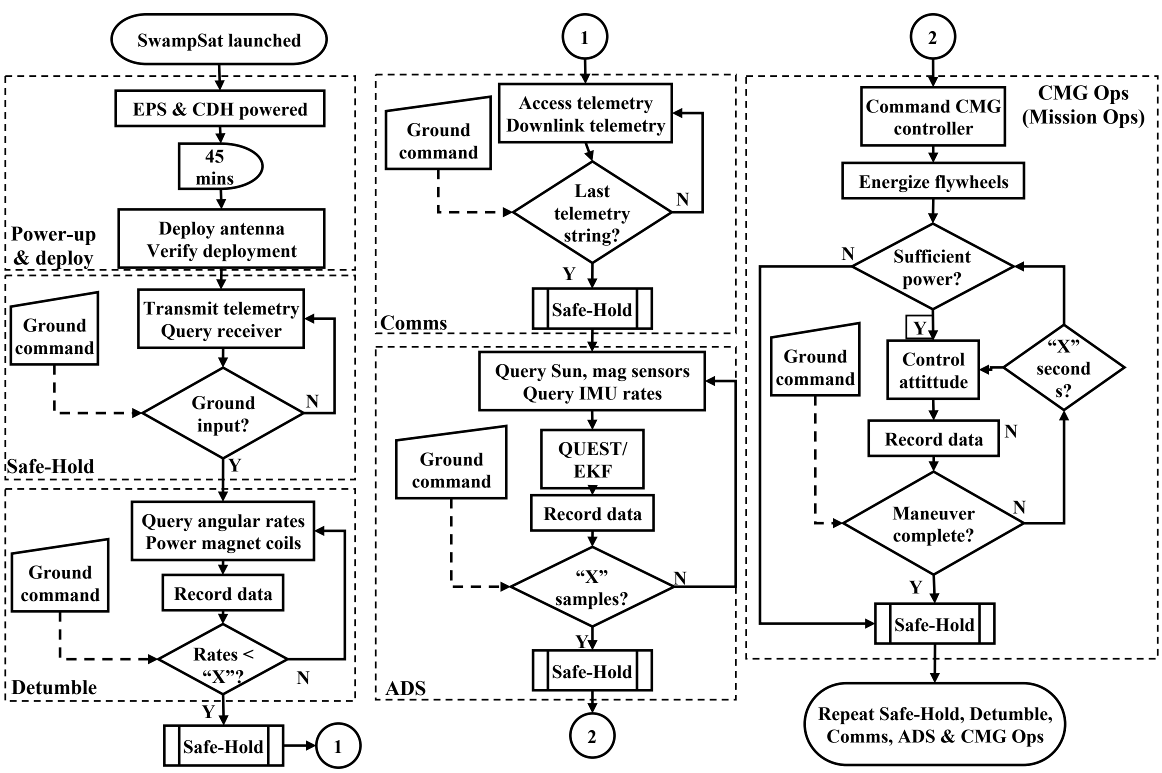


Figure 8. Mission CONOPS

up and deploy operation at the "Power ON" block shown in Fig. 9. The program communicates with the real time clock (RTC) and records the time as the current boot time on the EEPROM or flash storage. Along with the boot time, the boot counter is also updated on the flash storage. The boot counter is designed to track the number of times the satellite reboots due to insufficient power, ground command or due to a watchdog reset. Following the write process the satellite checks the launch flag and based on the status of the flag, it proceeds to either deploy the antennas or turn on the receiver. The antenna deployment, designed to be repeated if necessary, is verified by sensing the change in acceleration and/or satellite angular rates before and after the deployment. To protect the primary payload from any potential electromagnetic interference the satellite is designed to be idle for 45 minutes after P-POD launch. The launch flag is updated to indicate a successful deployment. After successfully executing the antenna deployment and the wait period the satellite enters safe-hold mode.

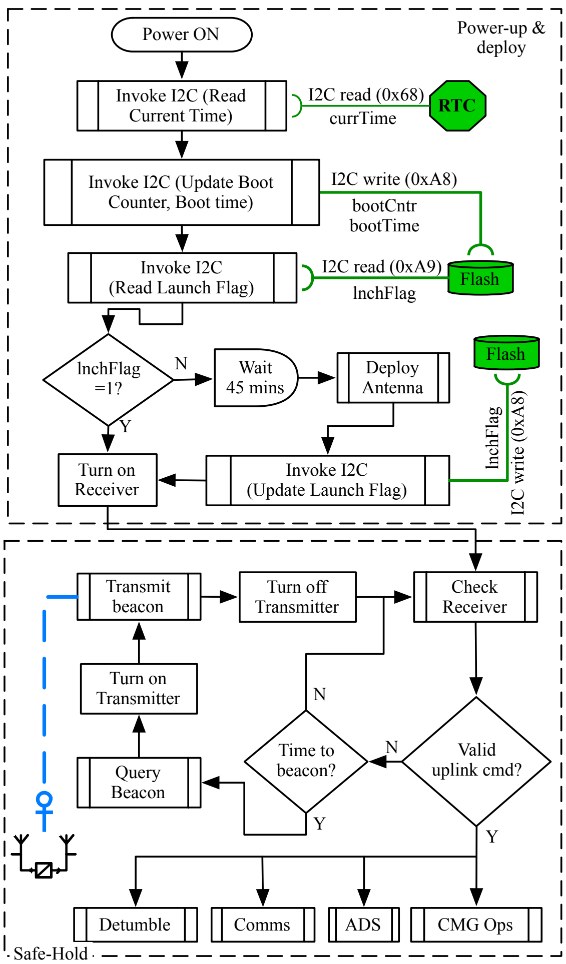


Figure 9. Power-Up & Deploy and Safe-Hold Modes

During safe-hold mode, the onboard transceiver is powered on and the satellite can be located by a ground station or a hand held operator. The safe-hold operating mode is a low power mode designed to validate secondary objectives and facilitate a net positive power generation from the solar cells. Real-time satellite health data is transmitted during this operating mode for ground-based decisions to switch to a particular operating mode. As indicated in the flowchart, safe-hold mode is designed as an infinite loop unless interrupted by a ground command or a power reset. The satellite receiver is in an interrupt mode and can be commanded by a ground station. While the satellite is listening for any ground communication via the receiver, the flight software collects real-time health data from the on board sensors and actively transmits it at specific intervals. As shown in Fig. 9, the satellite can be commanded into another operating mode through a ground input. It is important to note that the rectangular blocks represent tasks, which are derived from the mission mapping process. Similarly each color coded component and interface is also derived from the mission mapping process. The computational overhead associated with each task coupled with the duty cycle of its execution can facilitate the estimation of computational overhead of each operating mode and ultimately the entire flight software.

Safe-hold Mode Downlink Telemetry

Each operating mode is associated with a downlink telemetry, which is as stated earlier, formulated by combining the telemetry overheads associated with the components identified through the mission mapping process. The safe-hold mode downlink telemetry referred to as the satellite beacon is shown in Table 1. Within the limitations of the transceiver [25], the satellite beacon effectively relays its health to assist ground based decisions. The data transmitted as part of satellite beacon is shown in Table 1. As shown in the table the hardware column identifies the components derived from the mission mapping process and the bits column identifies the telemetry overhead associated with these components.

Safe-hold Mode Uplink Command

To affect parameters within each operating mode an uplink command is designed for each. The formulation of uplink command for each operating mode is influenced by the tasks, which are grouped to form the operating mode. The uplink command for the safe-hold mode and the specific parameters, which can be altered are shown in Table 2. The beacon interval, which is influenced by the query beacon task, is inversely proportional to the net power consumed by the onboard transceiver and this relationship is utilized for satellite power management. By making a provision for varying the beacon interval the satellite can be operated in a "super" safe-hold mode when the power being generated on board the satellite is at its minimum. The safe-hold mode uplink command makes provision to attempt re-deployment through the launch flag variable and makes provision for resetting satellite power. The deploy antenna and verify task influence this parameter within the operating mode.

8. CONCLUSION

Small satellites, particularly the pico- and nano-class CubeSats, are being recognized for their utility and as potential candidates for addressing the challenges of an operationally responsive space. While the ORS office seeks a new business model for meeting U.S. space capabilities, NASA's Franklin and Edison programs intend to mature technologies relevant for small satellites and support missions to prove these technologies. To address the challenges of the ORS office and NASA's Franklin and Edison programs it is critical to standardize satellite systems and reduce the uniqueness of each satellite design. The pico- and nano-class CubeSats could be capable of addressing these challenges but contrary to the design and development of traditional satellites, CubeSat designs have mostly been ad hoc. The approach adopted by traditional satellites may not be directly applied for the design

Table 1. Safe-hold Mode Downlink Telemetry

Quantity	Hardware	Interface	Bits	Hex Characters
time stamp				
year	RTC	I^2C	12	3
month	RTC	I^2C	4	1
date	RTC	I^2C	8	2
hour	RTC	I^2C	8	2
minute	RTC	I^2C	8	2
seconds	RTC	I^2C	8	2
MMDB1 temperature	TI-DSP	SPI	12	3
MMDB2 temperature	TI-DSP	SPI	12	3
SMDB1 temperature	TI-DSP	SPI	12	3
SMDB2 temperature	TI-DSP	SPI	12	3
IMU X	IMU	SPI	14	4
IMU Y	IMU	SPI	14	4
IMU Z	IMU	SPI	14	4
IMU temperature	IMU	SPI	12	3
battery voltage	EPS	I^2C	10	3
battery current	EPS	I^2C	10	3
battery bus current	EPS	I^2C	10	3
battery current direction	EPS	I^2C	10	3
battery temperature	EPS	I^2C	10	3
5v bus current	EPS	I^2C	10	3
3.3v bus current	EPS	I^2C	10	3
transmitter current	TCVR	I^2C	10	3
receiver current	TCVR	I^2C	10	3
boot count	Flash Storage	I^2C	12	3
boot time				
year	RTC	I^2C	12	3
month	RTC	I^2C	4	1
date	RTC	I^2C	8	2
hour	RTC	I^2C	8	2
minute	RTC	I^2C	8	2
seconds	RTC	I^2C	8	2
msp430 temperature	SFC430	A/D	12	3
		Total	**312**	**84**

Table 2. Safe-hold Mode Uplink Command

Description	Bits	Hex Chars	Value	Description
Operating mode ID	8	2	00	Safe-hold mode = 00
Beacon interval	8	2	XX (00 to FF)	00 corresponds to 30 s; XX corresponds to increments of 10 seconds; The frequency can be varied from 10 seconds to 2550 seconds (42 minutes)
Battery voltage	8	2	XX (00 to FF)	signed two's complement integer; 0 corresponds to 7.0v; each increment is 0.01 v
Launch flag	8	1	X (1 or 0)	Default flag is 0; Antennas can be attempted to be redeployed using the flag; flag == 1 indicates deployment should be repeated
Total	**32**	**7**	**00XXX**	

and development of these CubeSats. To address this issue, a pico- and nano-class CubeSat systems engineering approach, partially based on the guidelines of NASA's Systems Engineering Handbook, is described and presented.

The requirements flowdown is used to translate mission definition to satellite building blocks. The subsystems block, analogous to a coin-sorting machine, is a significant design element, which can be used to address COTS based development. To facilitate the estimation of mission overhead in terms of power, telemetry and computation, an overhead is associated with each building block derived from the mission mapping process. The components and interfaces derived from the mission mapping process can be used in an N2 diagram to facilitate the identification of conflicts early in the design phase. The tasks derived from the mission mapping process are grouped together to form operating modes. These operating modes form the building blocks of the flight software as layed out by the mission CONOPS. The computational overhead associated with tasks can be utilized to estimate the computational overhead of each operating mode and ultimately the flight software. Every operating mode is associated with a downlink telemetry and an uplink command, which when grouped together for all operating modes form the satellite telemetry budget. The telemetry overhead associated with the components facilitate the estimation of telemetry budget for each operating mode and the overall satellite. The components derived out of the mission mapping process can also facilitate the estimation of power budget, operating modes power budget, and mass budget.

REFERENCES

[1] H. Heidt, J. Puig-Suari, A. Moore, S. Nakasuka, and R. Twiggs, "Cubesat: A new generation of picosatellite for education and industry low-cost space experimentation," in *Proceedings of the Utah State University Small Satellite Conference, Logan, UT*. Citeseer, 2001, pp. 1–2.

[2] J. Schaffner and J. Puig-Suari, "The Electronic System Design, Analysis, Integration, and Construction of the Cal Poly State University CP1 CubeSat," in *16th AIAA/USU on Small Satellites Conference, Logan, UT*. Citeseer, 2002, pp. 1–2.

[3] B. Larsen, D. Klumpar, M. Wood, G. Hunyadi, S. Jepsen, and M. Obland, "Microcontroller design for the Montana Earth orbiting pico-explorer (MEROPE) CubeSat-class satellite," in *Aerospace Conference Proceedings, 2002. IEEE*, vol. 1. IEEE, 2005, p. 1.

[4] S. Waydo, D. Henry, and M. Campbell, "CubeSat design for LEO-based Earth science missions," in *Aerospace Conference Proceedings, 2002. IEEE*, vol. 1. IEEE, 2005, p. 1.

[5] M. Long, A. Lorenz, G. Rodgers, E. Tapio, G. Tran, K. Jackson, R. Twiggs, and T. Bleier, "A cubesat derived design for a unique academic research mission in earthquake signature detection," in *Proc. AIAA Small Satellite Conference*, 2002.

[6] C. Kitts, J. Hines, E. Agasid, A. Ricco, B. Yost, K. Ronzano, and J. Puig-Suar, "The GeneSat-1 Microsatellite Mission: A Challenge in Small Satellite Design," in *Proc 20th Annual AIAA/USU Conf on Small Satellites, Logan UT*, 2006.

[7] C. Kitts, K. Ronzano, R. Rasay, I. Mas, P. Williams, P. Mahacek, G. Minelli, J. Hines, E. Agasid, C. Friedericks *et al.*, "Flight Results from the GeneSat-1 Biological Microsatellite Mission," in *21 st Annual AIAA/USU Conference on Small Satellites*.

[8] C. Kitts, K. Ronzano, R. Rasay, I. Mas, J. Acain, M. Neumann, L. Bica, P. Mahacek, G. Minelli, E. Beck *et al.*, "Initial Flight Results from the PharmaSat Biological Microsatellite Mission," in *Proc. 23nd Annual AIAA/USU Conf on Small Satellites, Logan UT*, 2009.

[9] G. Minelli, A. Ricco, D. Squires, C. Beasley, and J. Hines, "O/oreos: A multi-payload technology demonstration," in *Proceedings of the 24th Annual AIAA/USU Conference on Small Satellites, Logan UT*, 2010.

[10] W. Bourne, "NRO CubeSat Experiments (Qb X)," 2009.

[11] H. Bahcivan, J. Cutler, J. Buonocore, and M. Bennett, "Radio Aurora Explorer: Mission overview and the science objectives," in *AGU Fall Meeting Abstracts*, vol. 1, 2009, p. 05.

[12] J. Klenzing, D. Rowland, J. Hill, and A. Weatherwax, "Firefly: A cubesat mission to study terrestrial gammaray flashes," in *AGU Fall Meeting Abstracts*, vol. 1, 2009, p. 1577.

[13] A. Jablonski, "Deorbiting of microsatellites in low earth orbit (leo)," 2008.

[14] NASA, *NASA Systems Engineering Handbook*. National Aeronautics and Space Administration, 2007.

[15] F. Leve, S. Allgeier, V. Nagabhushan, S. Asundi, D. Buckley, A. Waldrum, and T. Hiramatsu, "Astreci detailed design report, funsat iv design competition," University of Florida, 2007–2008.

[16] S. Allgeier, V. Nagabhushan, F. Leve, and N. Fitz-Coy, "Swampsat - a technology demonstrator for operational responsive space," in *Proceedings of CASI ASTRO 2010, 15th CASI (Canadian Aeronautics and Space Institute) Conference, Toronto, Canada*, 2010.

[17] S. Allgeier, V. Nagabhushan, N. Fitz-Coy, and W. Edmonson, "The role of universities in developing a responsive space industry," in *8th Responsive Space Conference, Los Angeles, CA*, March 8-11, 2010.

[18] NASA, "Office of the cheief technologist: Small spacecraft program," http://www.nasa.gov/pdf/474015main_SmallSat_Conference_8_9_10.pdf, 2010.

[19] F. Leve, "Development of the spacecraft orientation buoyancy experimental kiosk," Master's thesis, University of Florida, 2008.

[20] V. Nagabhushan, "Development of control moment gyroscopes for attitude control of small satellites," Master's thesis, University of Florida, 2009.

[21] S. Asundi, "Cubesat system design based on methodologies adopted for developing wireless robotic platform," Ph.D. dissertation, University of Florida, 2011.

[22] R. Nugent, R. Munakata, A. Chin, R. Coelho, and J. Puig-Suari, "The cubesat: The picosatellite standard for research and education," *Aerospace Engineering*, vol. 805, pp. 756–5087, 2008.

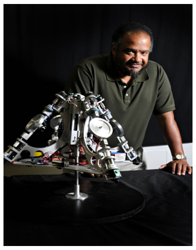
[23] J. Puig-Suari, C. Turner, and W. Ahlgren, "Development of the standard CubeSat deployer and a CubeSat class PicoSatellite," in *Aerospace Conference, 2001, IEEE Proceedings.*, vol. 1, 2001.

[24] R. Munkata, "Cubesat design specification rev. 12," August 2009.

[25] V. Galysh, "Katysat radio board, rev 2.4," January 2008.

BIOGRAPHY

Sharan Asundi is an Assistant Professor in the Department of Aerospace Science Engineering at Tuskegee University. He graduated with a MS and PhD in Aerospace Engineering from University of Florida in 2011, under the guidance of Dr. Norman Fitz-Coy. His research interests include design of autonomous ground and space systems, spacecraft attitude determination and estimation, vehicle health monitoring and design of small satellite mission operations.



Norman G. Fitz-Coy is an Associate Professor in the Department of Mechanical and Aerospace Engineering at the University of Florida in Gainesville. He is the Director of the Advanced Space Technologies Research and Engineering Center (ASTREC) - a National Science Foundation (NSF) Industry/University Cooperative Research Center. Dr. Fitz-Coy's research efforts address the dynamics and controls of multi-degree-of-freedom (M-DOF) systems with emphasis on space appications. One of the current research interests is attitude control of small satellites (less than 10 kg in mass). His research team has developed the world's smallest control moment gyroscopes and these will be flight verified the upcoming UF CubeSat mission SwampSat.