A Systems-Engineering Assessment of Multiple CubeSat Build Approaches

by

Zachary Scott Decker

B.S. Astronautical Engineering
United States Air Force Academy (2014)

Submitted to the Department of Aeronautics and Astronautics
in partial fulfillment of the requirements for the degree of

Master of Science in Aeronautics and Astronautics

at the

MASSACHUSETTS INSTITUTE OF TECHNOLOGY
June 2016

Author...
Department of Aeronautics and Astronautics
May 19, 2016

Certified by...
Kerri Cahoy
Assistant Professor of Aeronautics and Astronautics
Thesis Supervisor

Accepted by..
Paulo C. Lozano
Associate Professor of Aeronautics and Astronautics
Chair, Graduate Program Committee

Disclaimer: The views expressed in this thesis are those of the author and do not reflect the official policy or position of the United States Air Force, the United States Department of Defense, or the United States Government.

A Systems-Engineering Assessment of Multiple CubeSat Build Approaches

by

Zachary Scott Decker

Submitted to the Department of Aeronautics and Astronautics
on May 19, 2016, in partial fulfillment of the
requirements for the degree of
Master of Science in Aeronautics and Astronautics

Abstract

This research conducts a broad systems-based analysis of CubeSat engineering, with a focus on testing, failures, and their relationship to program cost, in order to assess multiple build approaches with a goal of maintaining the advantages of CubeSat missions while increasing reliability. In this work, the multiple approaches are called “beta build strategies,” and we show that satellite engineering groups with minimal experience can increase their probability of success by building two flight-model versions of their satellite, allowing for more exhaustive and potentially failure-inducing testing to be conducted on the first (beta version) satellite. This differentiates itself from the standard CubeSat build approach, which is typically to build a flat sat, then an engineering model, and then a flight model of the satellite. Frequently with CubeSat development, the additional expense of building a flight-like engineering model is avoided. However, in this work we consider the probability of success and overall cost impact for multiple approaches toward the flight build. We find that by spending an additional 33% of the planned program cost, a team which plans to take this alternate approach from the beginning can build and launch two flight-model versions of their spacecraft, increasing probability of success by 30%. This cost corresponds to a 40% saving from the scenario in which the decision to build a second flight-model spacecraft is made only after the first fails. The question which this analysis tries to answer is not, “how does a group spend the least amount of money to get their first CubeSat into space?” but rather, “how does a group spend the least amount of money to get a CubeSat into space that works?”

Thesis Supervisor: Kerri Cahoy
Title: Assistant Professor of Aeronautics and Astronautics

Acknowledgements

This research was conducted with support of both the MIT Lincoln Laboratory and MIT Space Systems Laboratory. The author would like to thank Gunter Krebs and Dr. Michael Swartwout, the individuals behind Gunter's Space Page [1] and the St. Louis University's CubeSat Database [2] , as these databases provided much of the historical spacecraft data used in this thesis. The author would also like to thank the sponsors of the Microwave Radiometer Technology Acceleration (MiRaTA) project, the NASA Earth Science Technology Office (ESTO), for providing the environment in which the space systems engineering knowledge necessary for this thesis could be learned. Finally, the author would like to thank his family, friends, and colleagues for their support, as well as his advisor, Kerri Cahoy, for her ongoing inspiration and mentorship.

Table of Contents

List of Figures

List of Tables

List of Acronyms

ADCS – Attitude Determination and Control System
BIB – Bottom Interface Board
BPI – Bus-Payload Integration
CDR – Critical Design Review
CG – Center of Gravity
CSK – CubeSat Kit
DAQ – Data Acquisition
EM – Engineering Model
EMC – Electromagnetic Compatibility (Testing)
EMI – Electromagnetic Interference (Testing)
EOM – End of Mission
GCR – Galactic Cosmic Radiation
GPSRO – Global Positioning Satellite Radio Occultation
GSE – Ground Support Equipment
I&T – Integration and Testing
ISS – International Space Station
LEO – Low Earth Orbit
LISP – Launch Integration Services Provider
LVLH – Local Vertical/ Local Horizontal
MicroMAS – Micro-sized Microwave Atmospheric Satellite
MiRaTA – Microwave Radiometer Technology Acceleration (CubeSat)
NASA – National Aeronautics and Space Administration
NDA – Non-Disclosure Agreement
NOAA – National Oceanic and Atmospheric Administration
PDM – Power Distribution Module
PDR – Preliminary Design Review
PIM – Payload Interface Module
PO – Purchase Order
PSR – Pre-shipment Review
S/C – Spacecraft
SPE – Solar Particle Event
SRR – System Requirements Review
SV – Space Vehicle
RBFP – Remove Before Flight Pin
RFP – Request For Purchase
TIB – Top Interface Board
TKD – Thermal Knife Driver

Chapter 1: Introduction and Motivation

1.1 Introduction

Once an activity in which only large, federally-funded organizations could participate, satellite engineering now has many smaller, private sector companies and university research group participants. The constraints typical to satellite engineering projects become even more pronounced in these small groups, where limited time, money, and human resources heavily shape how the project is undertaken. In order to address these challenges, these groups have adopted a new philosophy in the engineering of satellites, adopting commercially available parts, limited testing, and a higher risk posture that permits multiple attempts at success.

As the industry changes, seeing the addition of university and private sector participants, so too does the strategy of satellite engineering. Mission architectures of these smaller groups typically employ multiple smaller platform satellites, which leverage redundancy and reduced per-unit cost to allow for higher risk acceptance. The catalyst for this strategy of satellite engineering was the introduction of the “CubeSat” – a common form factor standard for miniature satellites, which has greatly increased the accessibility of the satellite engineering industry and has brought about a new era of miniaturized space technology applications [3].

This heightened risk-taking strategy has not come without shortcomings: over half of all successfully launched CubeSats to date have not met all of their primary mission objectives [2]. Compared with large-scale satellite projects, whose launch is planned in advance to coincide with the completion of the spacecraft’s build and integration, CubeSat projects, which rely on secondary payload opportunities for launch, are often not notified of their launch opportunity until as late as their equivalent Critical Design Review (CDR), or midway through the project. This can lead to accepting a launch date which is earlier than when the project would be able to deliver their spacecraft with the desired amount of testing and characterization completed. In addition, it is starting become more common to outsource much of the subsystem build to external vendors who provide commercial off-the-shelf (COTS) components, though delays in deliveries of these components are typical. The launch uncertainty and uncertainty in vendor delivery times and quality of subsystems leads to compressed timelines and shortcutting or eliminating some of the necessary pre-delivery actions (like testing) – which may increase the likelihood of failure.

This thesis analyzes the relationships between testing, failures, and the cost of CubeSat programs in order to offer insights into how the engineering strategies behind these projects can be improved. This analysis is then used to motivate multiple different approaches to build strategies for entry-level satellite engineering programs in order to yield greater success rates. These strategies, which we will call the CubeSat "beta build" strategies, attempt to show that satellite engineering groups with minimal experience can increase their probability of success by building two flight-model versions of their satellite, allowing for more exhaustive and potentially failure-inducing testing to be conducted on the first (beta version) satellite, with minimal impact to the program cost and schedule. This differentiates itself from the standard CubeSat build approach, which is typically to build a flat sat, then an engineering model, and then a flight model of the satellite. Frequently with CubeSat development, the additional expense of building a flight-like engineering model is avoided. However, in this work we consider the probability of success and overall cost impact for multiple approaches toward the flight build. The structure and cost of the "beta build" mission architecture are analyzed with representative testing and build parameters and are compared to traditional approaches to satellite engineering for various programs. The question which this analysis tries to answer is not, "how does a group spend the least amount of money to get their first CubeSat into space?" but rather, "how does a group spend the least amount of money to get a CubeSat into space that works?" The goal of this research is to offer a design and build strategy for CubeSats which maintains the advantages of CubeSat missions (decreased cost, shorter build times, and distributed mission architectures) but mitigates associated high risk and probable failures.

1.2 CubeSats

From the largest man-made satellite, the International Space Station, which has an approximate mass of 450,000 kg and is the size of a football field, to the currently known smallest, the KickSat Sprite, which have a mass of only a couple grams and the size of a large postage stamp, satellites come in many different shapes and sizes [4] [5]. Because of the diversity of satellites seen today, a number of standardized satellite classes, based on the spacecraft's mass, time to build, and cost, have been created (Table 1-1). It should be noted that the values given in this table for cost and time of development are only estimates and, based largely on the complexity of the spacecraft's payload, can vary quite substantially across programs.

Table 1-1: Satellite Size Classifications [6]

Type	Mass (kg)	Cost (US $)	Time of Development from Proposal to Launch
Large Satellite	>1,000	0.1-2 B	>5 years
Medium Satellite	500-1,000	50-100 M	4 years
Mini-satellite	100-500	10-50 M	3 years
Micro-satellite	10-100	2-10 M	~1 year
Nanosatellite	1-10	0.2-2 M	~1 year
Pico-satellite	<1	20-200 K	<1 year
Femto-satellite	<0.1	0.1-20 K	<1 year

A subclass of nanosatellites, called the CubeSat, has seen rapidly growing popularity. Created by Jordi Puig-Suari and Bob Twiggs in 1999, the CubeSat standard provides the industry with a common platform for compact satellites and defines a safe container for these compact satellites, which launch service providers feel is low risk to their primary payload and are willing to accept on their launch vehicles [3] [7]. Their creation was the solution to the problem that the duration of satellite engineering projects greatly exceeded the duration of a graduate student's academic curriculum; with the CubeSat, a graduate student could feasibly design, build, test, and even operate a satellite within his/her time as a student [8].

CubeSats are among the smallest operational satellites, with masses ranging from roughly 1 kg to 14 kg. CubeSats are sized by unit ("U"), with 1U corresponding to a size of 10 cm x 10 cm x 10 cm and a mass of 1.33 kg. A 3U CubeSat, to provide an example, has the size and mass of three 1U blocks placed on top of one another: 10 cm x 10 cm x 30 cm with a mass of about 4 kg. CubeSats commonly have 1U and 3U form factors because current deployers are capable of deploying multiples of 1U up to 3U. A CubeSat deployer is a metal structure surrounding the CubeSat with an opening hatch and a spring ejection system. Common 3U deployers used today are the P-POD, NanoRacks, and ISIPOD deployers [3] [9] [10]. The universal form factor of a CubeSat is one of its greatest advantages, enabling these standardized deployers to be easily integrated into launch vehicles as auxiliary or secondary payload opportunities. This gives CubeSats access to many more launch opportunities than other satellite form factors. Table 1-2 provides a list of current launch providers and secondary payload accommodations for CubeSats.

Table 1-2: CubeSat Launch Providers [11] [12] [13] [14] [15] [16] [17] [18]

Launch Vehicle	Provider	Launch Sites	Payload Accommodation
Atlas V, Delta IV	NASA/ULA, USAF	CCAFS[a], VAFB[b]	Optional EELV[c] secondary payload adapter (maximum 24 CubeSats planned)
Delta II	NASA (ELaNa[d])	VAFB	3 P-PODs demonstrated
Dnepr	ISC Kosmotras	Baikonur[e]	5 P-PODs demonstrated
Falcon I	SpaceX	Kwajalein[f], CCAFS	Ride Share Adapter: 6 P-PODs maximum
Minotaur I	USAF (OSC[g])	VAFB, Wallops FF[h]	1 or 2 P-PODs per launch
Minotaur IV	USAF (OSC)	Kodiak[i], VAFB	Maximum 4 P-PODs planned
Neptune 30	IOS Inc.[j]	Eua Isle, Tonga	Maximum 4 1U CubeSats
PSLV[k], GSLV[l]	ISRO[m]	Satish Dhawan SC[n]	P-POD, number not yet available
Taurus XL	NASA (OSC)	VAFB, CCAFS, WFF, Kwajalein	Maximum 3 P-PODs planned
Vega	ESA	Kourou, Fr. Guina	Maximum 3 P-PODs planned
Firefly α	Firefly Space Systems	TBD	Dedicated CubeSat launch – 400 kg total
Electron	Rocket Lab	Mahia Peninsula[o]	Dedicated CubeSat launch – 150 kg total
LauncherOne	Virgin Galactic	Various[p]	Dedicated CubeSat launch – 200 kg total

[a] Cape Canaveral Air Force Station, Florida, USA.
[b] Vandenberg Air Force Base, California, USA.
[c] Evolved Expendable Launch Vehicle.
[d] Educational Launch of Nanosatellites program.
[e] Also known as Tyuratam, located in Republic of Kazakhstan.
[f] SpaceX launch facility on Omelek Island in Kwajalein Atoll, Republic of the Marshall Islands.
[g] Orbital Sciences Corp.
[h] NASA Wallops Flight Facility Wallops Island, Virginia, USA.
[i] Kodiak Launch Complex, Kodiak Island, Alaska, USA.
[j] Interorbital Systems.
[k] Polar Satellite Launch Vehicle.
[l] Geosynchronous Space Launch Vehicle.
[m] Indian Space Research Organization.
[n] Satish Dhawan Space Center.
[o] New Zealand Mahia Peninsula
[p] Launches from Boeing 747-400 carrier aircraft

Since the first mission in 2000, the utilization of CubeSats as a space mission platform has increased rapidly (Figure 1-1). By the end of 2015, over four hundred CubeSats have launched – three-fourths of which were launched in the last three years. The greatest percentage of CubeSats launched have been of the 3U form factor (53%), followed closely behind by 1U CubeSats (31%). Note that “Opal,” as seen in Figure 1-1, was a preliminary CubeSat form factor design created at Stanford University’s Space Systems Development Laboratory [19]. As the number of CubeSats being utilized grows, so does the number of applications. Table 1-3 provides example CubeSat missions in the many science and technology fields in which they are already utilized.

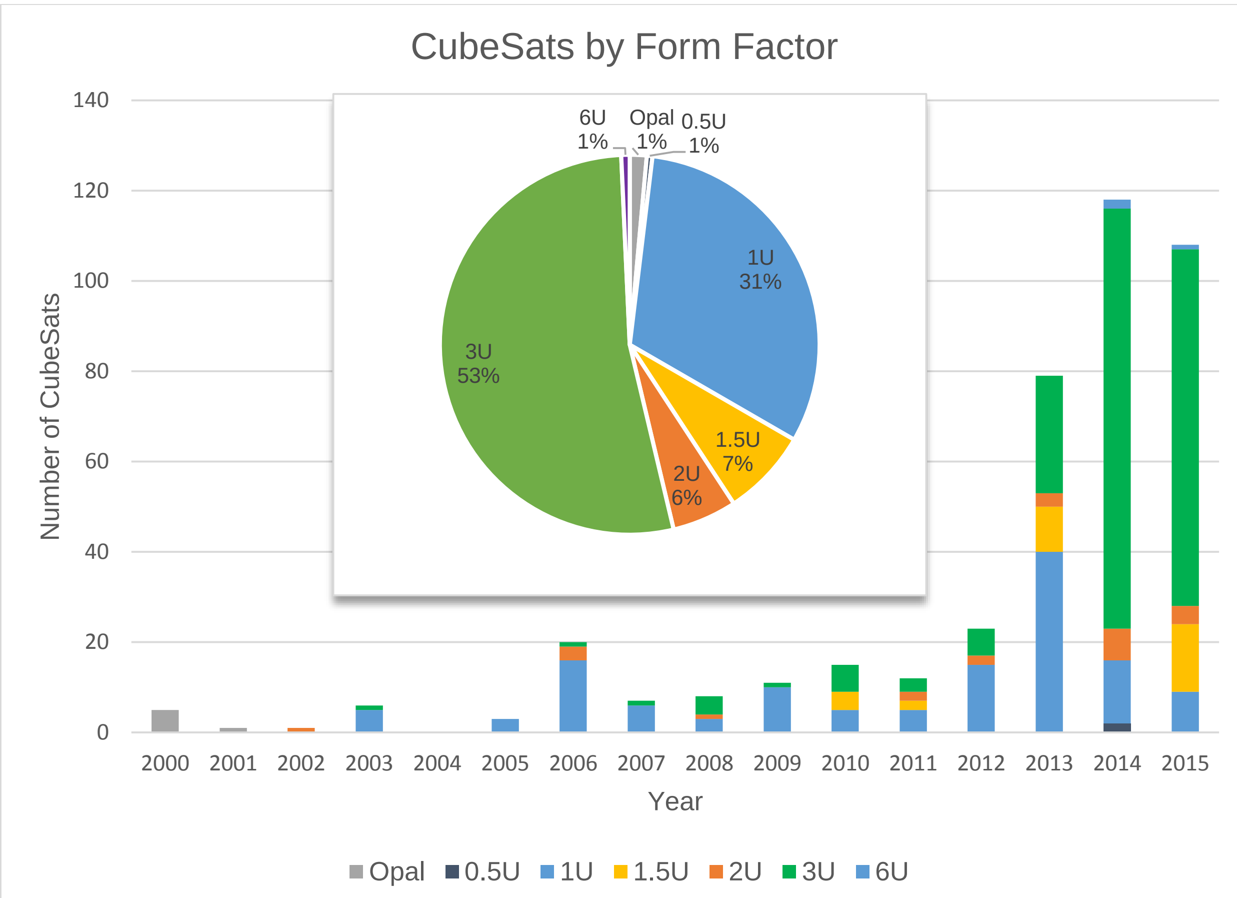
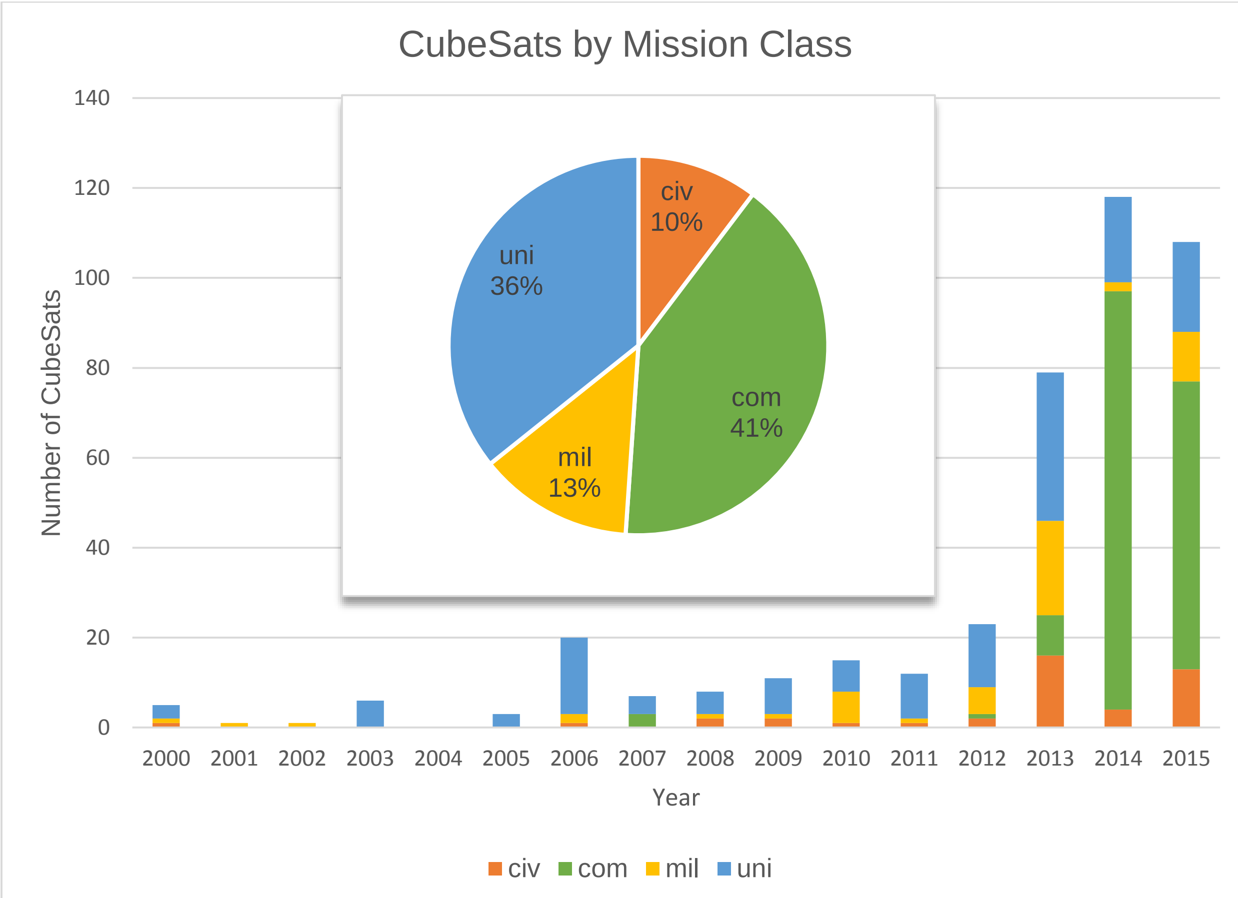


Figure 1-1: CubeSats by Form Factor [1] [2]

Table 1-3: Example CubeSat Applications [11]

Field	Mission	Sponsor/Lead Agency
Astrobiology	**O/OREOs**: UV-visible spectral monitoring: organic materials; space radiation effects on survival/ growth of 2 microbes.	NASA/ARC
Astronomy	**BRITE/ CanX-3/ TUGSAT-1**: Constellation of nanosatellites for asteroseismology.	CSA/U. Vienna/Austrian Research Promotion Agency (FFG)
Atmospheric Science	**FIREFLY**: Terrestrial gamma-ray flashes induced by lightning.	NSF
Biology	**GeneSat-1**: E. coli gene expression via fluorescent reporters in microgravity.	NASA/ARC, Stanford University, Santa Clara University
Earth Observation	**PRISM**: Validation of medium-resolution earth observation	University of Tokyo
Ecology	**NCube2**: Large ship automatic identification system; reinder tracking	Norwegian University of Science and Technology
Electronics	**Robusta**: Validate test standards for space radiation impact on electronics	CNES/ESA/Montpellier University
Materials Processing	**HawkSat 1**: Commercial materials processing research	Hawk Institute for Space Sciences
Pharmaceuticals	**PharmaSat**: Antifungal agent dose response of yeast in microgravity	NASA/ARC, U. Texas Medical Branch, Santa Clara University
Technology Demonstration	**MAST**: Electromagnetic tether technology demonstration	Tethers Unlimited
Space Weather	**CINEMA**: detection of sub-atomic particles from space magnetic storms	UC Berkeley Space Sciences Lab/ Imperial College/ NSF
Telecommunications	**NEMESIS**: survey spectrum 1-1300 MHz: document radio-frequency interference	US Naval Academy

CubeSats are significant because they are making the satellite industry more accessible. Once an endeavor in which only large military and civilian government organizations could participate, satellite engineering now has numerous other participants, including university and commercial groups. In fact, within the realm of CubeSats, university and commercial groups now make up the majority of the field (Figure 1-2).



civ: Civilian government organizations (e.g. NASA, JAXA, ESA).
com: Commercial, private organizations.
mil: A government military or defense organization (e.g. U.S. Air Force).
uni: A university or other educational institution.

Figure 1-2: CubeSats by Mission Class [1] [2]

Prior to 2014, the university groups dominated the CubeSat arena, at that point in time making up nearly 60% of CubeSat missions. Commercial groups which utilize CubeSats have since surpassed university groups. Namely, the San Francisco, California-based Planet Labs, for example, launched four of their LEO Earth-imaging “Doves” in 2013 and then launched 83 in 2014 [2] [20].

CubeSats are useful for research and technology demonstration because they provide a unique platform in which the traditional methods of satellite engineering can be modified and/or accelerated. This is due to a number of factors, namely: (1) CubeSats often utilize commercial off-the-shelf components and the systems are generally less complex, which drives down development time and per-unit cost; (2)

CubeSats conform to specific, standardized form factors, decreasing the amount of trade space available for overall spacecraft design and also making it possible to use common types of deployers that are low-risk to launch service providers, supporting many more launch opportunities; (3) Due to their lower per-unit cost and shorter build times, CubeSats have a much higher risk acceptance level than traditional satellite engineering projects, allowing for certain testing and integration strategies to be altered, accelerated, or disregarded (4) CubeSats are changing the way in which the industry sees satellites: by utilizing them as expendable and rapidly replaceable assets within constellation mission architectures, CubeSat groups can achieve greater global and temporal coverage and shorter revisit intervals, which can complement, if not replace, large monolithic systems.

1.3 Scope of Thesis

This thesis attempts to provide a broad systems engineering-based overview of CubeSat engineering, with a focus on aspects of testing, failures, and their relationships with total program cost. Experience from two MIT-affiliated CubeSat projects, the Microsized Microwave Atmospheric Satellite (MicroMAS) and the Microwave Radiometer Technology Acceleration satellite (MiRaTA), is leveraged in this research, with examples and data referenced from both. We next provide a brief overview of these two satellites [21] [22].

1.3.1 MicroMAS

The Microsized Microwave Atmospheric Satellite (MicroMAS) is the first in a line of joint MIT and MIT Lincoln Laboratory low-Earth orbiting atmospheric weather sensing CubeSats [21] [23]. The goal of MicroMAS is to serve as a technology demonstration mission for microwave radiometers onboard a CubeSat platform performing atmospheric characterization. The idea for these satellites is to have progressive increases in complexity and capability, with the long-term goal of having a constellation of atmospheric sensing 3U CubeSats as part of the TROPICS project by the year 2020 [24].

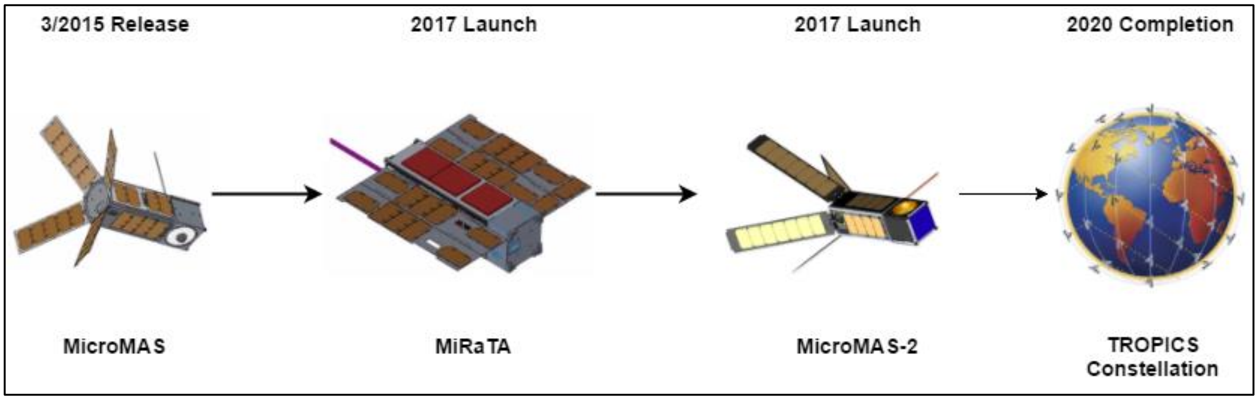


Figure 1-3: MIT Atmospheric-Sensing CubeSat Timeline

One key aspect of the MicroMAS project is that it was a joint effort between a government organization and a university group: government Lincoln Laboratory employees were in-charge of the build and test of the payload while graduate students at MIT's Space Systems Laboratory were responsible for the design and build of the spacecraft bus, with MIT Lincoln Laboratory mentoring and support. MicroMAS is a dual-spinning spacecraft, with the payload being a 1U-standalone microwave radiometer unit which rotates approximately once per second, with the rest of the spacecraft LVLH stabilized (local vertical local horizontal) during primary mission operations [25]. Such a design gives a clear distinction between the payload portion of the spacecraft and the remaining bus portion.

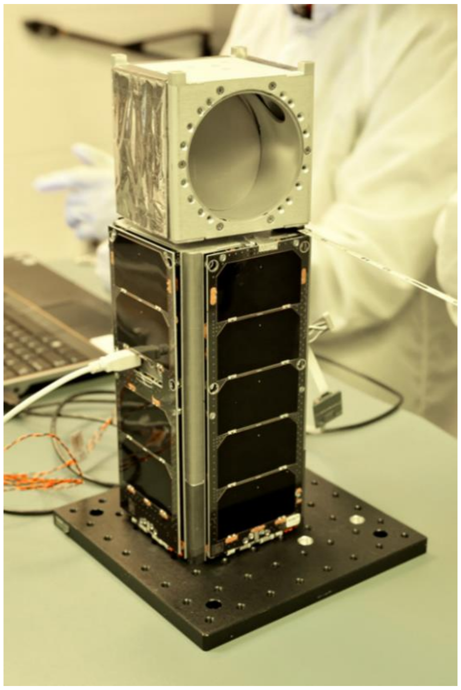


Figure 1-4: MicroMAS 3U CubeSat [25]

The microwave radiometer for MicroMAS is single-band, at approximately the 118-GHz oxygen absorption line, and detects microwave emissions in the upper atmosphere to create atmospheric temperature maps which aid in extreme weather forecasting. The radiometer spins at a rate of approximately once per second; when facing towards Earth, it sweeps a large area of atmosphere in its 2.5 degree beamwidth and 20 km diameter footprint. MicroMAS uses the cold temperature it measures when pointed at deep space as a cold calibration point [26].

The initial plan was to have MicroMAS prove the single-band miniaturized microwave radiometer technology and then have the follow-on project, MiRaTA, demonstrate a multi-band radiometer with GPS radio occultation calibration shortly thereafter. Unfortunately, MicroMAS failed to achieve all of its primary mission objectives due to a loss of contact-failure (this is discussed further in Section 3.2), which prompted the decision to re-fly MicroMAS as the MicroMAS-2 CubeSat, which would be built concurrently with MiRaTA and would launch around the same time as MiRaTA. MicroMAS-2 is nearly identical to MicroMAS, with the exception that it uses a next-generation, tri-band radiometer [27].

1.3.2 MiRaTA

The Microwave Radiometer Technology Acceleration (MiRaTA) 3U CubeSat is a NASA ESTO sponsored CubeSat that was selected as part of the Educational Launch of Nanosatellites (ELaNa) 14 Mission. It is MIT's successor to the MicroMAS project and has a primary mission objective to demonstrate a tri-band radiometer which measures upper atmospheric temperature, humidity, and cloud ice, and is then calibrated by GPS radio occultation (GPSRO) measurements using a secondary patch antenna payload, the Compact TEC (Total Electron Count) and Atmospheric GPS Sensor (CTAGS) [22] [28] [29]. This information is used to improve weather forecasting on Earth and will advance the NASA Technology Readiness Level (TRL) from 5 to 7 for both the radiometer and GPSRO payloads. MiRaTA is funded by the NASA Earth Science Technology Office (ESTO) and is another collaborative-build CubeSat with the following organizations taking part: MIT Space Systems Laboratory (spacecraft bus), MIT Lincoln Laboratory and University of Massachusetts Amherst (radiometer), Space Dynamics Laboratory (ground support), and The Aerospace Corporation (GPSRO).

During MiRaTA's primary mission phase, the goal is to scan the same patch of atmosphere with both the radiometer and GPSRO payloads. Nominally the MiRaTA spacecraft is local-vertical local-horizontal (LVLH) stabilized, with the nadir face pointing towards Earth. Once data acquisition commences, the nadir-facing radiometer scans a patch of atmosphere and then MiRaTA performs a pitch-up maneuver to expose the formerly zenith-facing GPS patch antenna to the same patch of atmosphere that the radiometer scanned. The GPS patch antenna then collects data from any GPS satellites during ingress occultations (as they move behind the Earth from the perspective of MiRaTA), after which MiRaTA pitches back down to resume its LVLH orbit [30].

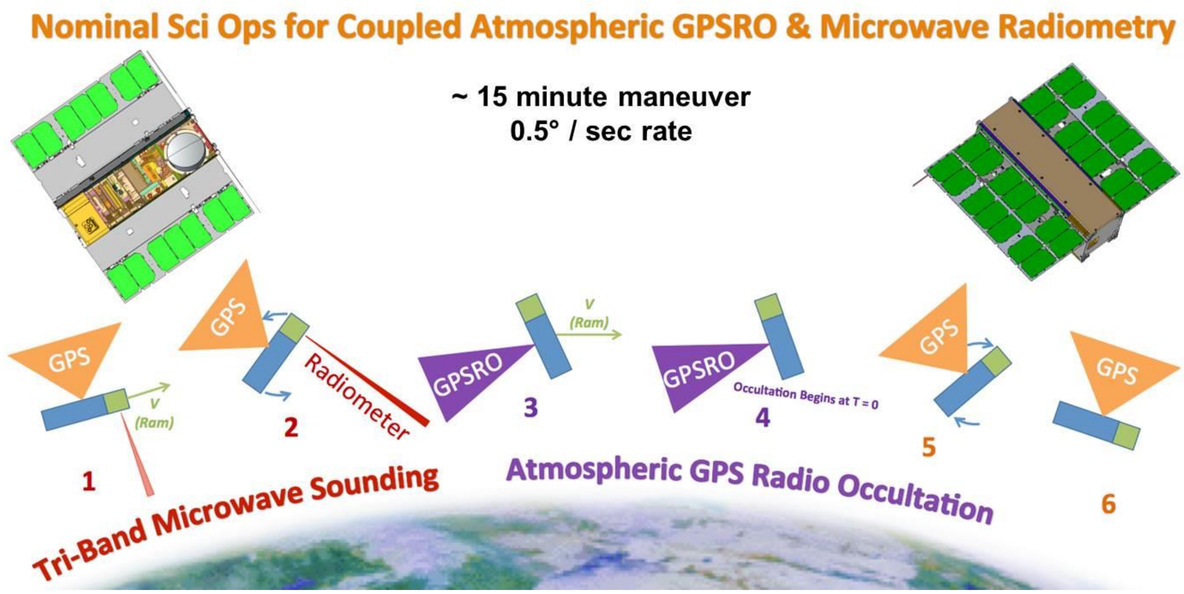


Figure 1-5: MiRaTA Mission Operations [31]

The requirements derived from the primary mission objectives drive the design of the MiRaTA spacecraft. MiRaTA gathers approximately 141 MiB per day with three maneuvers and needs to be able to transmit this data to the ground station quickly and efficiently. This leads to a stringent power requirement on the spacecraft (12 W orbit average power), which drove the decision to use an array of double-deployed solar panels in order to generate enough power. The solar panels, in addition to the battery and Electrical Power System (EPS), were purchased as COTS components from a CubeSat vendor. The payload volume of the MiRaTA spacecraft increased from that of MicroMAS, from approximately 1U to just under 2U, while the required capabilities of the bus increased (need for increased memory storage and the incorporation of a backup radio on the motherboard). In order to meet these requirements, many design decisions were made to make the avionics subsystem more compact (by removing unnecessary stock components and combining normally separate components onto a single board) [26]. Based on the nature of the pitch-up maneuver, the radiometer payload remains fixed, with field of view pointing down to Earth from the nadir face of the spacecraft, while the GPS patch antenna payload takes up the zenith face of the spacecraft. The Attitude Determination Control System (ADCS), which controls this maneuver and maintains the spacecraft orientation during all other phases of flight, has most of its components housed in an assembly that is located in the anti-ram end of the spacecraft. The MiRaTA spacecraft design is shown in Figure 1-6 and Figure 1-7.

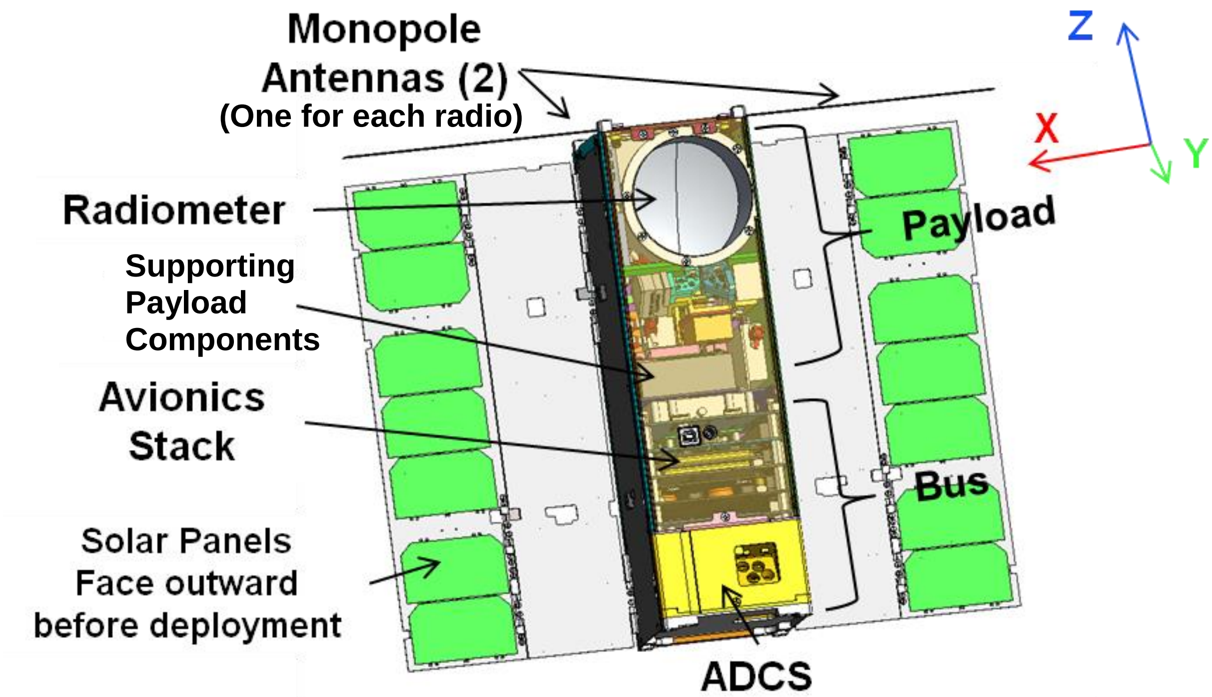


Figure 1-6: MiRaTA Design Overview, Front Face [31]

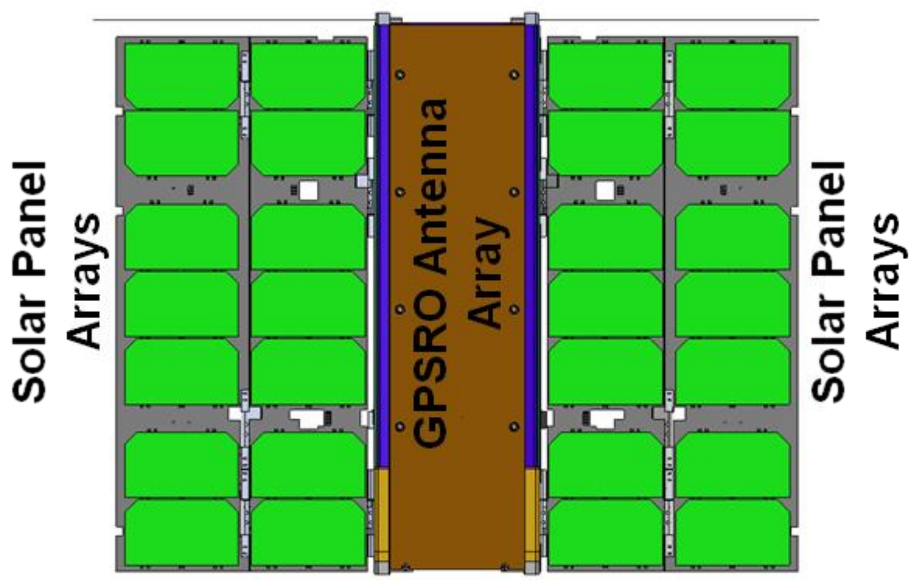


Figure 1-7: MiRaTA Design Overview, Back Face [31]

This thesis leverages much of the work from the MiRaTA project to serve as an example CubeSat engineering program. The high demands and relative complexity of this satellite, when coupled with the heavy resource constraints common to CubeSat projects, resulted in a need for an efficient testing and integration strategy. Limited funds meant engineering model components (duplicates prior to flight specifically for testing) could not be purchased for every subsystem. Limited human resources, in addition to time constraints, meant any testing that would be conducted needed to be well-planned out, testing multiple components or subsystems simultaneously, while still maintaining a high level of oversight in order to avoid accidental mishaps. In addition, being the successor to a project which was unable to achieve its mission objectives due to an on-orbit failure, there is extra emphasis on the MiRaTA project to verify proper integrated space vehicle functionality prior to flight. These aspects of the MiRaTA mission set the context for the motivation behind this thesis: trying to reduce risk and increase probability of success in a CubeSat program while still maintaining its relative low-cost and quick-build advantages.

1.3.3 Organization

The goal of this work is to provide insight into how the engineering strategies behind CubeSat engineering projects can be improved. In Chapter 2, the details of preflight testing on a CubeSat is discussed, using details from the MiRaTA project as an example. Chapter 3 focuses on failures across CubeSat programs to-date, with the intent to identify primary sources and modes of failure. In Chapter 4, the total program cost for a CubeSat is analyzed for various attempt to success (attempt:success) ratios in order to identify the major cost drivers for a project and help derive different build strategies for CubeSats which decrease risk while minimizing subsequent cost increase. Based on the findings in Chapter 4, Chapter 5 discusses three “beta build” strategies in which CubeSat engineering groups build two versions of their satellite, allowing for more exhaustive and potentially failure-inducing testing to be conducted on the first (beta version) satellite, and compares the cost and structure of programs utilizing these strategies to those utilizing traditional strategies.

Chapter 2: CubeSat Testing

2.1 Introduction to CubeSat Testing

Testing is important for any engineering project, but especially for satellites due to the nature of their operational environment, space, which limits the ability to conduct repairs when there is an on-orbit fault or failure. Tests on a satellite program verify that a component, subassembly, or the integrated space vehicle performs as expected, and continues to do so under its operational conditions. They also serve to validate requirements and verify robustness of components and their interfaces, ensuring spacecraft survivability through testing and launch vehicle integration, launch, and on-orbit operations.

2.2 Planning for Testing – Systems Engineering

Testing on a satellite program is a time-consuming process, both in the amount of time it takes to actually complete the tests and also the time required to adequately prepare for, and analyze the data from, tests. Usually, a satellite project's launch integration services provider (LISP) will require that a specific set of tests are completed prior to spacecraft delivery. For a CubeSat, which will be an auxiliary or secondary payload, the primary purpose of these required environmental tests from the perspective of the LISP is to verify that the CubeSat will not break apart or cause debris in any way which could damage the primary payload or launch vehicle. From the perspective of the CubeSat team, these tests should demonstrate that the CubeSat can meet its mission objectives. Fitting all of the environmental tests within the compressed timeline of a CubeSat project can be challenging, and advanced planning is necessary.

Systems engineering is a broad definition for the particular type of engineering concerned with both the high-level and subsystem development and interactions of a program and includes: 1) compatibility between subsystems and proper functionality of integrated assemblies; 2) development and management of the schedule, manpower, and cost required to deliver completed subsystems as well as to integrate and deliver the space vehicle; 3) ensuring that subsystem testing validates requirements, and planning for and conducting integrated testing following subsystem delivery; 4) analyzing program risks throughout in order to identify and implement mitigation strategies to ensure on-time delivery. Systems engineering-based thinking enables a program to adequately plan for and complete all required tests prior to spacecraft delivery.

One systems engineering tool which is often used on satellite projects is a “”testing and integration flow diagram.” These diagrams may be used at different assembly levels, for example, one for space vehicle testing following bus and payload delivery, one each for the bus and payload testing and integration strategy, and sometimes for the more complex subsystems as well. Figure 2-1 provides an example space vehicle testing and integration flow diagram. Note that this diagram starts with the integration of the completed spacecraft bus and payload then details the required environmental tests and steps leading to spacecraft delivery.

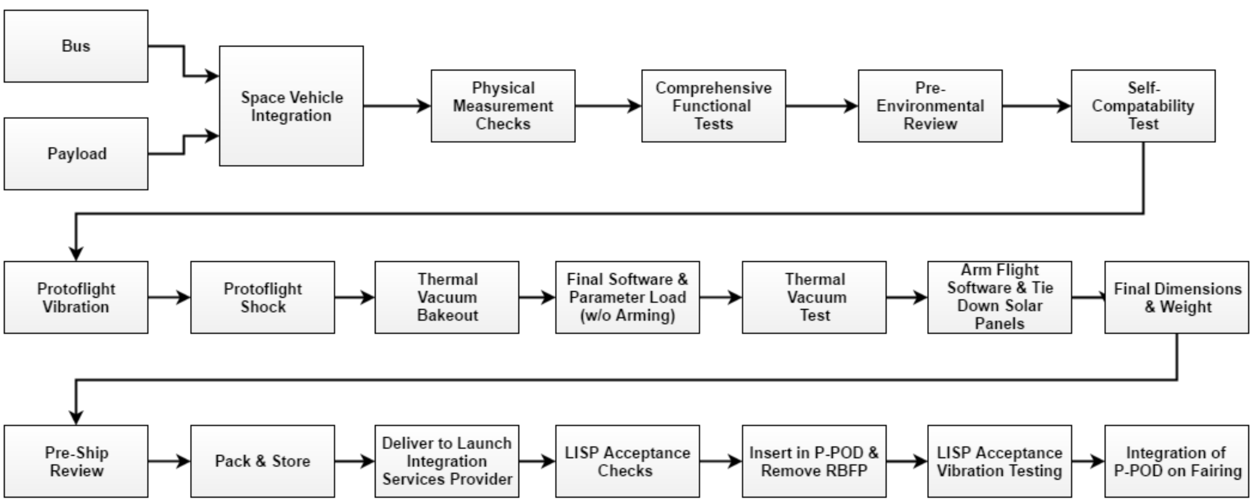


Figure 2-1: Space Vehicle Testing and Integration Flow Diagram

This testing and integration flow diagram describes the general order in which the many tests and milestones for the space vehicle will be completed in order to aid in planning and prioritizing the team’s efforts during the final months of the satellite project. The dates when these tests occur, as well as their duration, are typically tracked in the program’s master schedule in order to ensure other prerequisite items, such as ground support equipment design and fabrication, are met in time for tests.

Such integration and test planning is conducted at the lower bus and payload system levels as well. Typically these diagrams have more detail than the higher-level space vehicle diagram as they include information on all the subsystems, as well as lower-level functional testing and requirement verification. Figure 2-2 provides an example bus integration and testing flow diagram. In this diagram, subsystem activities are tracked on a per-month basis, with each block time-stamped with the amount of months remaining until bus-payload integration (BPI). These flow diagrams serve to aid as tools in the testing and integration of the spacecraft and, as such, are dynamic documents which change as the program progresses (such as after component deliveries are delayed or when testing discovers that rework is required).

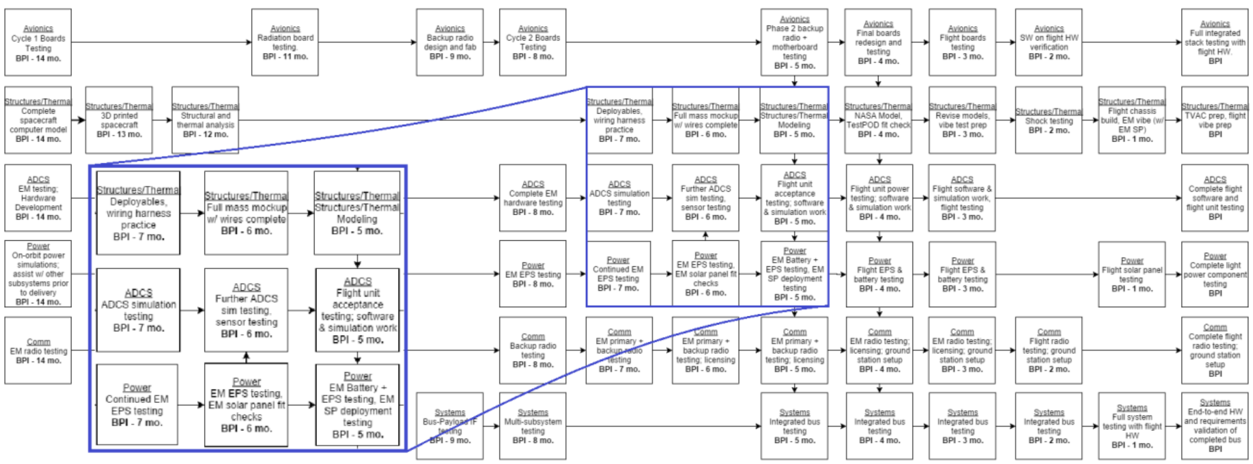


Figure 2-2: Bus Integration and Testing Flow Diagram

The next step before testing is to plan for the actual tests themselves. These plans are usually in the form of procedures which provide the engineers conducting the test with the following information: 1) pictures and descriptions of the initial set-up and component configuration which needs to be met prior to test initiation; 2) step-by-step instructions of actions to be taken and areas of caution which require extra attention; 3) reporting instructions and details for requirements which are to be verified through the test. In addition, the systems engineer or team management must determine and implement a standard reporting format for the tests (usually in the form of a “logbook”) prior to the tests being conducted, as properly documenting the actions during and results of tests is critical to an engineering project. This enables continuity between team members, allows for progress to be tracked, and provides team members with information on hardware in the event of a mishap, fault, or failure so that solutions and steps forward can be determined. However the systems engineer wishes to implement these logbooks, be it online or through physical logbooks which follow the hardware, may depend on the team structure and number of facilities being used. In the case of MiRaTA, which very frequently has team members at separate locations, an online logbook on a secure server with restricted access was the best solution so that all team members could read test reports simultaneously regardless of their location. Another supplementary tool which was added on the MiRaTA project due to its multiple-location aspect was a "Hardware Tracker," which tracks the location of engineering and flight unit components and documents any changes made to hardware – serving as both a location and "current physical state" tracker for the spacecraft hardware. With all of these tools implemented to date, the MiRaTA project has been able to successfully plan for, conduct, and analyze the results of all necessary tests within its short program timeline.

2.3 Test Levels

Satellite tests take place over the course of a program at a number of different levels. These levels are differentiated such that spacecraft components and assemblies are tested at varying degrees of intensity based on their stage of development. Generally speaking, early versions of components (like engineering units) will be tested much more exhaustively than the final versions which will be used for flight. The philosophy behind this is to determine the full extent of the testing article's capabilities, operational bounds, and performance on early models in order to prevent overstressing later flight units which then need only limited verification of proper functionality. The different testing levels are defined below [32] [33]:

- Development tests – conducted on engineering components (representative articles used strictly for test, not for flight) to collect data and validate the design approach.
- Qualification tests – conducted on engineering components to verify design requirements are met. Margin values and product robustness are also verified.
- Protoqualification (also known as protoflight) tests – conducted on test components which are limited in production and could be used for flight (usually

called protoflight hardware), these tests verify design requirements using reduced amplitude and duration margins.

- Acceptance tests – testing that flight hardware is free of workmanship defects, meets specified performance requirements, and is acceptable for delivery.
- Pre-launch validation tests – conducted at the launch base to ensure hardware, software, and personnel support is ready for the launch and mission.
- Post-launch validation tests - serve as the operational verification of requirements and are used to monitor and detect faults over the spacecraft's lifetime.

2.4 Types of Tests

Across the testing levels, tests can be further defined by their type. This research divides the analysis of testing seen on CubeSat projects into two separate groups: functional testing and environmental testing, the latter of which comprises vacuum, thermal, vibration, shock, and radiation testing.

2.4.1 Functional Testing

Functional testing is the verification that a particular component does what it was designed to do. For this research, functional testing is further defined to include all tests that analyze the performance of a component both individually and as a part of an integrated assembly. Performance and interface testing, as well as electromagnetic interference and compatibility testing (EMI and EMC), tests common to the satellite engineering field, are included within this definition of functional testing.

Functional testing can be both mechanical and electrical; examples include deployment testing of solar panels (mechanical) and measuring the correct voltages coming out of a power distribution unit (electrical). Functional testing takes place during each testing category; engineering components are functionally tested during development and qualification or protoqualification testing and flight components are functionally tested during final acceptance testing. Functional testing takes place on the component, subassembly, and integrated space vehicle levels, making it by far the most fundamental, as well as the most frequently conducted, testing on satellite programs. The remainder of this section describes some of the functional testing conducted on the

Microwave Radiometer Technology Acceleration (MiRaTA) 3U CubeSat to serve as a representative example of testing performed on CubeSats. For organization purposes, these functional tests have been further divided into two categories: those which are electrical, and those which are mechanical, in nature.

2.4.1.1 Functional Testing (Electrical)

As was mentioned earlier, functional testing occurs during all levels of testing over the course of a program. The primary difference between qualification and protoqualification (protoflight) tests is that for qualification tests, it is a given that the components being tested will not be used on flight, while protoqualification tests are conducted on components that may be used for flight (usually called “protoflight” components). For MiRaTA, protoqualification testing occurred on the Electrical Power System (EPS). In this situation, the team received two identical EPS units from the vendor, with both units being rated for flight. Since both units could potentially be used for flight, it would be inappropriate to conduct qualification-level testing on these components (the vendor does their own qualification testing first, then builds these to flight, so we can assumes some level of qualification testing has been completed). The team therefore decided that one unit should be designated the protoflight unit, which would receive the brunt of the testing, while the other would be set aside for flight. Doing this protects one of the EPS units from being damaged by improper testing – a possibility that, no matter how many precautions may be taken to avoid, may still occur. Testing would need to be conducted on the flight EPS unit regardless, but by the time this occurs, the team would have gained experience with the specifics of testing on the first EPS, reducing the risk of human error on the second unit. This is not to say the first unit could not be used for flight, however: if after all testing was complete and the results showed that the first unit actually performed better (and no damage occurred during testing), it would be designated the flight unit. Figure 2-3 shows a functional protoqualification test conducted on the EPS unit.

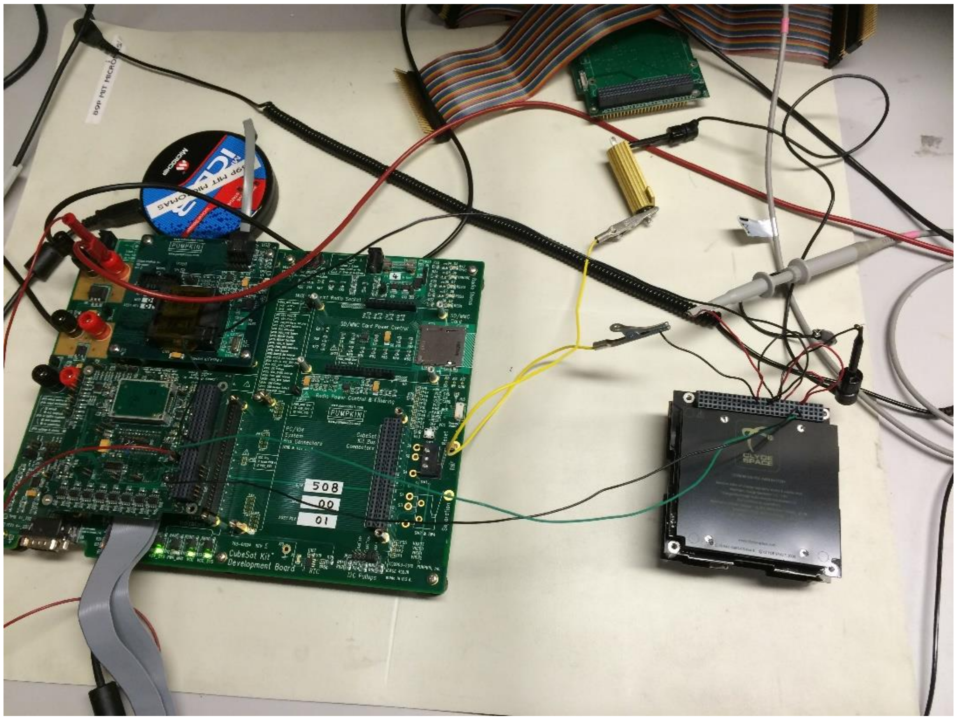


Figure 2-3: Functional Protoqualification Testing – MiRaTA EPS PDM Reset Setup

In this test, the EPS Power Distribution Modules (PDMs) reset command is tested. The EPS is powered by the battery, placed on top, and is connected to a development board through an I^2C Data and I^2C Clock line routed into the avionics stack's Bottom Interface Board (BIB). The BIB and motherboard are connected via traces on the development board. When commanded, the PDM units should bring specified voltage lines down to zero for approximately 600 ms, and then back up to the proper voltage. The 5 V EPS line is sampled with an oscilloscope when the command is sent, and the proper response is observed, validating the proper functionality of the EPS PDM reset command, as well as the interface path between the spacecraft microcontroller, BIB, and EPS (Figure 2-4).

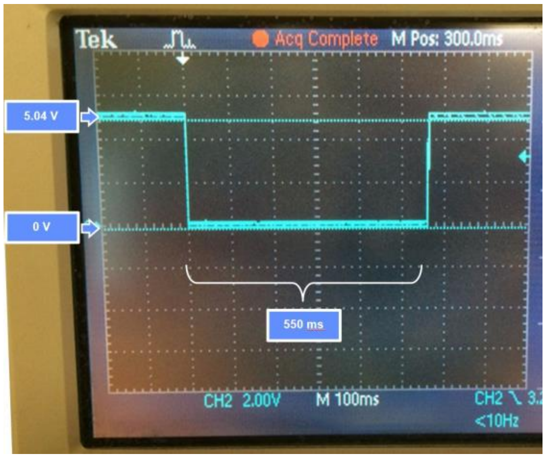


Figure 2-4: Functional Protoqualification Testing – MiRaTA EPS PDM Reset Results

The next functional test example is conducted on the spacecraft's ADCS unit, testing its functionality as part of an integrated subassembly. Protoflight units of the stack's Top Interface Board (TIB), Bottom Interface Board (BIB), and microcontroller are plugged into the development board. The Attitude Determination and Control System (ADCS) is then integrated via a connection in the BIB and the development board is connected to a computer, which communicates with the spacecraft's microcontroller to send and receive telemetry and commands. Figure 2-5 below shows this setup.

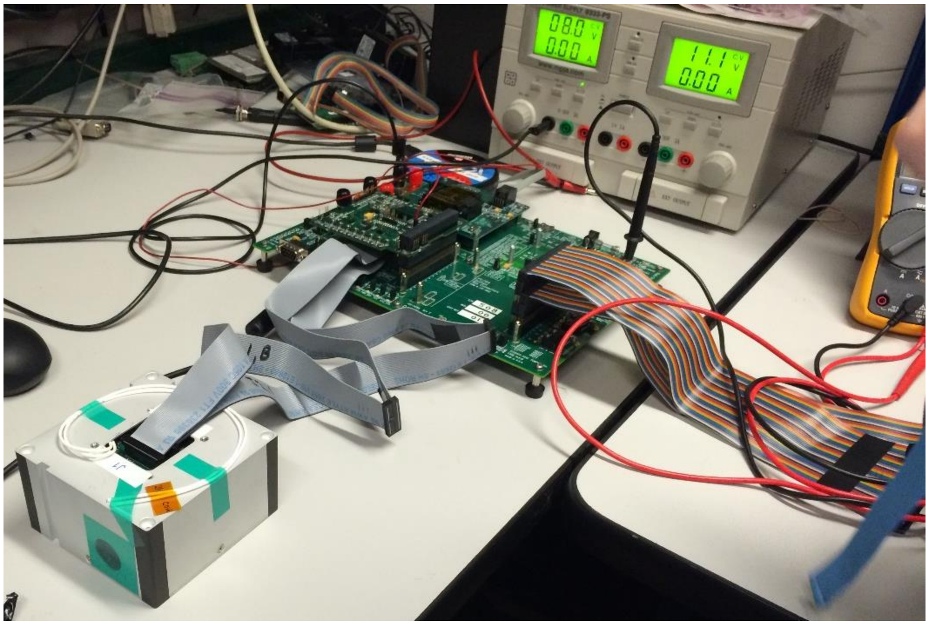


Figure 2-5: Functional Protoqualification Testing – MiRaTA ADCS Subassembly

The purpose of this test is to verify the proper functionality of, and communication line between, these avionics boards and the ADCS unit. Commands are sent by the user through the computer's command module to the spacecraft microcontroller, which then passes the command through the BIB and into the ADCS unit to spin up one or a combination of wheels. Information about the wheel rates is then sent back to the command module through the same path. Not only does this verify the design of the ADCS unit, but it also demonstrates the proper functionality of the microcontroller, BIB, and the interfaces between them.

During the final stages of a program, functional testing is conducted at the acceptance level. At this point, all units being tested could be viable options for flight, so the utmost care and precautionary measures must be taken. The following example is of a simple, but very important, test: the proper interface between flight components of the bus stack. Each constituent board has approximately the same length and width dimensions (approximately 97 mm x 97 mm), are separated by standard-length hexagonal aluminum standoffs, and interface both physically and electrically through two 52-pin CubeSat Kit (CSK) Bus Headers (Figure 2-6). For this particular test, the flight unit of the primary radio had not yet completed its own component testing so it was not integrated into the rest of the stack. This test, though seemingly rudimentary, was actually quite the milestone for this project. First, every board was able to connect with one another – and without any interference or clearance issues between components on the top and bottom of the boards (which could be a serious concern). Next, the proper electrical connection between the boards, as well as the proper traces on the boards themselves, were verified. The battery provides the power source to the EPS, which then converts the raw power into specific, regulated voltage lines and then distributes the power to the TIB, BIB, Motherboard, and the primary and backup radios. Though the only visual result of the electrical connection within this test were blinking status lights on the motherboard, it represented quite the success as it was the culmination of many different components of the bus all working, and for the first time, together.

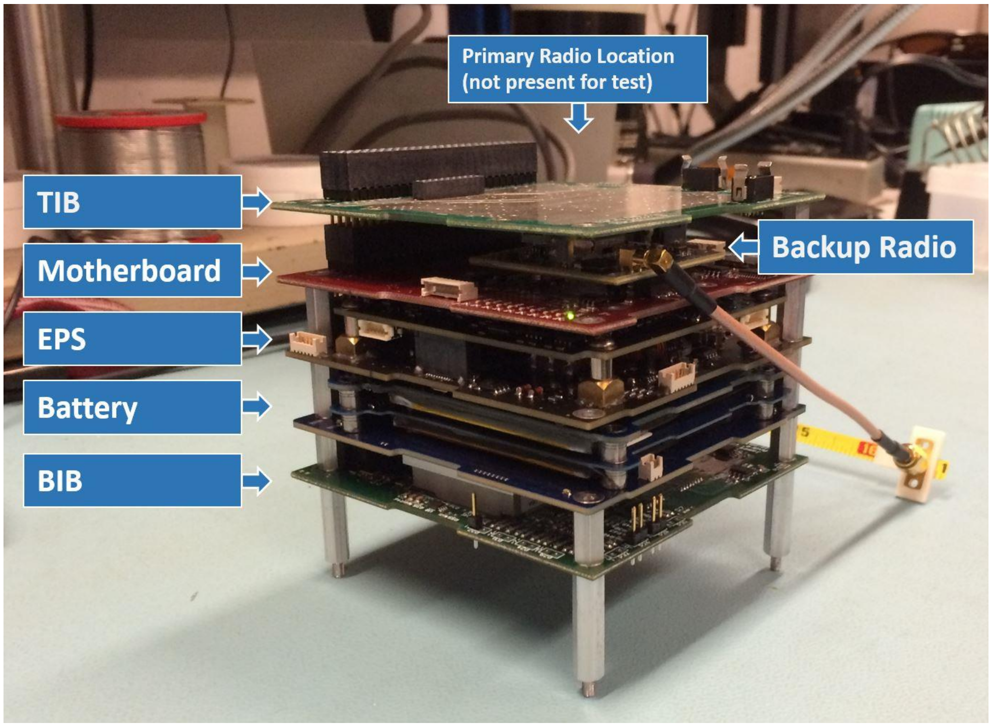


Figure 2-6: Functional Acceptance Testing – MiRaTA Bus Stack

2.4.1.2 Functional Testing (Mechanical)

Functional testing on a CubeSat project does not always involve circuit boards and other electrical components; mechanical functional testing is also necessary. Three mechanical functional tests will be described here: fit-checks, cable routing, and deployment testing.

First, fit-checks verify that components fit within specified constraint requirements and that the assembled space vehicle properly fits within the deployer. One of the most challenging aspects of a CubeSat project is keeping everything within a very constrained volume – strict requirements with very tight tolerances are placed on CubeSat teams for the spacecraft dimensions so that the CubeSat tracks along the deployer rails, protrusions are within clearance requirements, and inhibit switches are depressed when stowed. These requirements ensure that the spacecraft exits safely during deployment and remains powered off until then as well.

The first fit-check described here is that of the solar panels. The constraints on the dimensions of solar panels are extremely tight, especially for the MiRaTA project. For MiRaTA, the solar panels needed to be a very precise length and width, such that potential cell area was maximized using a double-deployable configuration while still fitting the panels within the designated recess on the spacecraft chassis. In addition, the solar panels needed to be less than 6.5 mm thick when stowed in order to meet the launch integration services provider's protrusion requirements for the deployer. Figure 2-7 shows the fit-check of the engineering solar panels for MiRaTA. This test turned out to be critical to the project: while the solar panels met the length and width dimensions, fitting within the chassis recess, bowing on the panels caused them to be out of specification on their 6.5 mm thickness requirement when stowed. For this reason, the solar panels were returned to the vendor for rework and design changes to ensure the flight units would meet specifications. Following these changes, the solar panels met all required design specifications, fitting within the designated recess while being less than 6.5 mm thick at all points along the chassis when stowed.

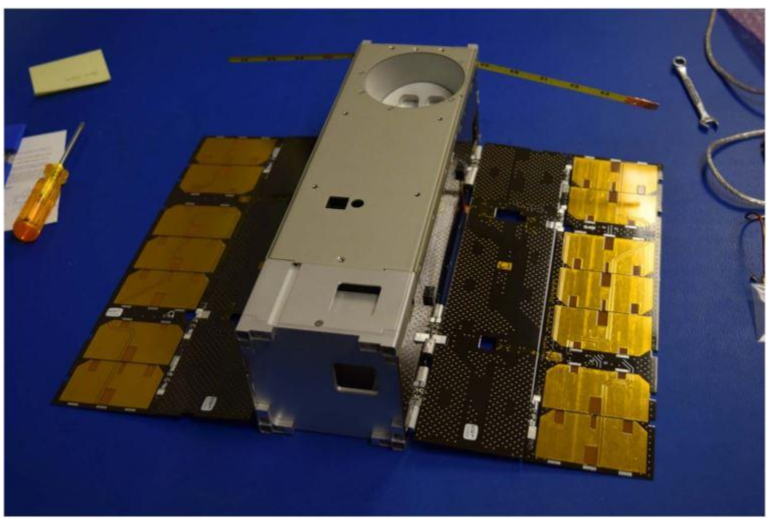


Figure 2-7: MiRaTA EM Solar Panel Fit Check

Next, Figure 2-8 shows the fit check of the MiRaTA mass mockup with its deployer. The MiRaTA mass mockup uses engineering model avionics components with representative masses of equal dimension and center of gravity (CG) for internal payload components and a structural chassis equivalent to flight to provide an accurate representation of the assembled space vehicle. It is used to verify that all components fit within the 3U CubeSat chassis and that the assembled vehicle then integrates properly with the deployer. For MiRaTA's mission, a Cal Poly P-POD deployer is used, so a representative "TestPOD" was supplied by the launch integration services provider and used for this fit check. Note that the solar panels are not present in this picture as they had already been returned to the vendor for reworks at this point, though the clearances from the space which they would occupy, as well as the proper fit of the rest of the space vehicle, was still verified through this test.

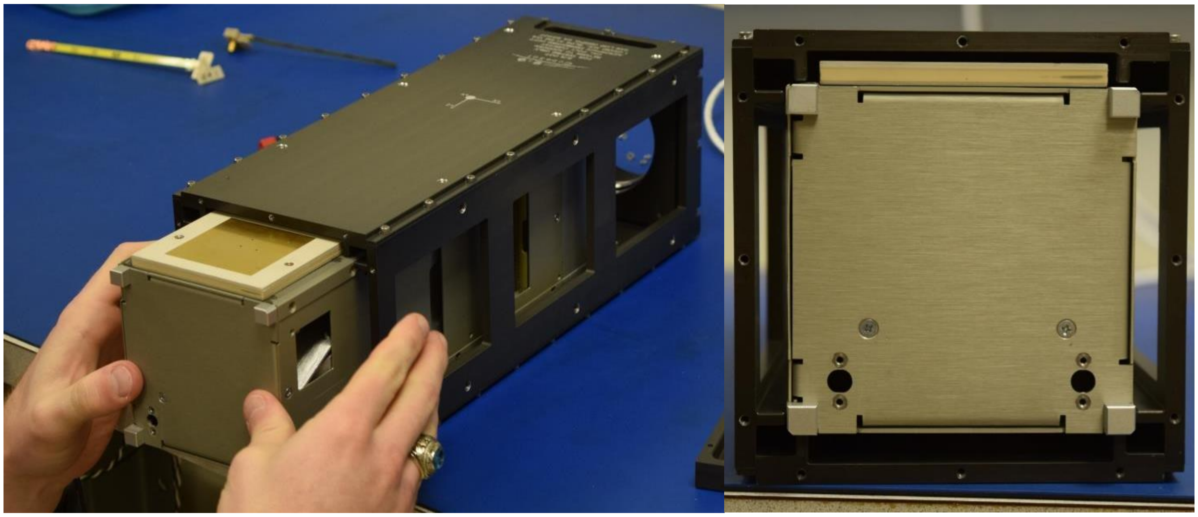


Figure 2-8: MiRaTA Mass Mockup + TestPOD Fit Check

Another example of a common mechanical functional test is cable routing. Wires and cables (the two terms used here interchangeably) are a major concern for any CubeSat engineering project, as they all need to fit within the very limited spaces between the internal components of the spacecraft. On multiple occasions for the MiRaTA project, it was discovered that components needed to be moved or support components needed to be completely redesigned only after cable routing was conducted and it was revealed that the cables would not fit in the current spacecraft configuration. Figure 2-9 shows the routing of the primary radio coax cable through the mass mockup. This figure emphasizes the challenge of cable routing in a CubeSat: while the straight-line distance between the start and termination points is only about 3 inches, the length of the cable needed to be 13 inches because of the complex path it has to take to be able to fit, snaking back and forth through both sides of the spacecraft. In addition, conducting this test revealed an area which could have been a major concern if it were discovered too late: the connector on the end of the coax cable was too long; when completely screwed onto the radio, the end of the connector protrudes past where the side panel of the chassis would go. This initiated a reorder of cable assemblies with shorter connectors in order to be within specifications.

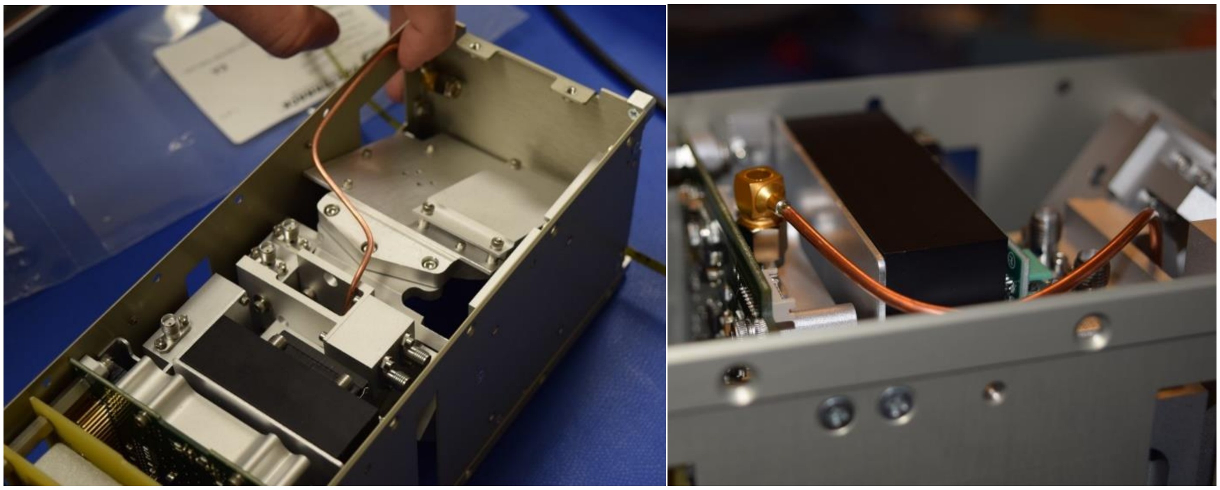


Figure 2-9: MiRaTA Cable Routing

The final mechanical functional test example is deployment testing. This test ensures that the solar panels successfully transition from the stowed to deployed configuration when commanded to do so. For MiRaTA, there were three goals for the deployment testing: 1) train personnel to tie proper stowing knots, 2) verify hardware works properly when cold, and 3) verify hardware works properly in a space-simulated environment.

The solar panels deploy by the spring force in their hinges once the tie-down cord is cut. These cords are hand-tied by the team engineers and keep the solar panels in their stowed configuration until a command is sent to the spacecraft to fire thermal knife drivers (TKDs), which are sources of electrical current which heat up and burn through the cords. Current is applied to all cords on each set of solar panels simultaneously, with the goal of symmetrical panel deployment. In addition, an extra set of TKDs exist on each panel assembly for redundancy in the event of a failed TKD firing.

In order to verify successful solar panel deployment, a test rig was designed to hold the mass mockup in a vertical orientation (gravity neutral for the solar panels). Figure 2-10 shows this setup. A solar panel assembly is mounted to an aluminum wall which is fastened to a black-anodized aluminum base plate. A black-anodized aluminum shroud is then fastened to this base plate and covers the entire structure. The testing rig is then placed inside of a bell jar to achieve vacuum. The purpose of the base plate and shroud is to cool the entire testing rig to the required steady state temperature prior to the deployment actuation. Since the structure of the testing rig obscures the view of the deployment, sensors are placed inside the rig to detect the deployment of the panels: photodiodes detect laser light when the panels are stowed and then, upon deployment, the panels obscure the lasers which cuts off the light to the sensors. In addition, accelerometers are placed on the panels themselves to detect the forces of deployment experienced on the panels.

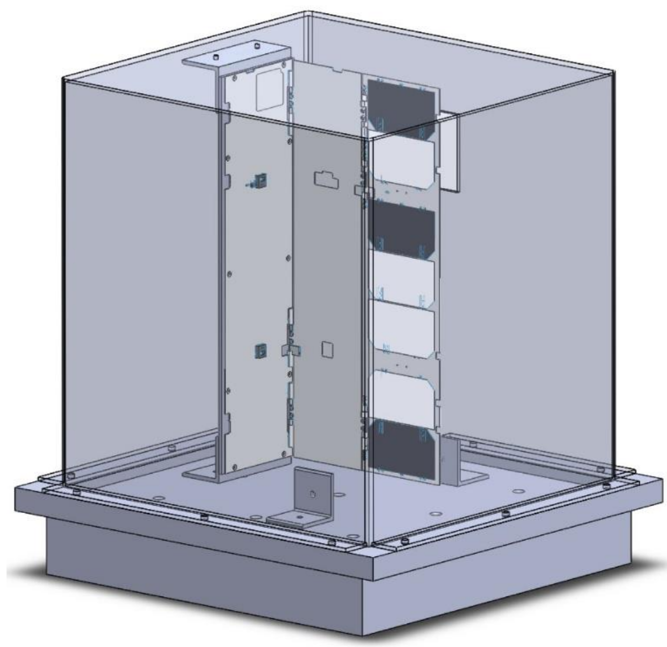


Figure 2-10: MiRaTA Solar Panel Deployment Testing Rig

2.4.2 Environmental Testing

Environmental testing includes all actions taken to quantify the spacecraft's response to, and ability to survive, the environmental conditions seen during launch and its orbital lifetime. These tests include vibration, shock, thermal, vacuum, and radiation testing.

2.4.2.1 Thermal and Vacuum Testing

Thermal and vacuum testing attempt to simulate the extremes of the on-orbit environment which the space vehicle will face. The spacecraft's thermal environment is dictated by its orbit and the power output and subsequent heat generation of its components. Each component has a survival temperature range (the lowest and highest temperatures it can experience in order to avoid damage) and an operational temperature range (the lowest and highest temperature values between which the component can operate). Because satellites are in the vacuum of space, heat cannot be distributed between components through convection. This necessitates the use of active thermal management tools on the spacecraft; in order to keep all components within their operational temperature ranges, the following tools are used: thermal sensors, which monitor subsystem and component temperature, thermal straps, which route heat away from hot components, and heaters, which provide heat to the cold components.

The testing schemes for thermal and vacuum testing on a satellite program typically involve qualification or protoqualification thermal testing on individual components (e.g. the radio, a payload, etc.) early in the program, with protoqualification and acceptance thermal vacuum testing on the integrated space vehicle at the end of the program. An additional thermal vacuum bakeout takes place just prior to delivery to remove all contaminants from the spacecraft prior to integration. The goal of thermal vacuum testing is to verify and validate thermal models, operational software and algorithms, data packaging and downlink, and to identify workmanship errors. Additionally, thermal vacuum testing can aid in developing ground segment processing.

On MiRaTA, thermal vacuum testing will be conducted during three different testing regimes. The first is to be conducted on the payload prior to integration to verify payload functionality during thermal vacuum conditions. Similarly, the integrated space vehicle will be subject to two cycles of thermal vacuum testing, one prior to delivery to the launch integration services provider and one following delivery for bakeout. This will verify the proper functionality of the integrated spacecraft during the environmental extremes it will face on orbit and will give the LISP sufficient information to verify that there will be no outgassing concerns with the spacecraft prior to deployment.

2.4.2.2 Radiation Testing

Another aspect of the space environment which should be accounted for is radiation. Radiation in space is most commonly in the form of ionizing high-energy charged particles. When these charged particles interact with spacecraft, they have the potential to cause disturbances such as bit flips and single-event upsets which can lead to faults within the spacecraft electronics and failures. The sources of these charged particles are trapped radiation, galactic cosmic radiation (GCR), and solar particle events (SPE) [34]. Of the greatest concern to low Earth orbit (LEO) satellites is trapped radiation, which are charged particles that are captured by, and then travel along, the Earth's magnetic field lines. There are two primary "belts" of space around Earth where these charged particles are concentrated, which are referred to as the Van Allen radiation belts. The inner belt contains primarily protons while the outer belt contains primarily electrons, the reason for this distribution being the particles' masses. There are also high energy particles that tend to precipitate along the magnetic field lines at the poles, and in an area which is known as the South Atlantic Anomaly, which are of the greatest concern to low Earth orbiting (LEO) satellites, whose orbits may cross through these areas, exposing the satellite to the charged particles therein. Additional sources of radiation are galactic cosmic radiation and solar particle events. Galactic cosmic radiation consists of ionized atoms originating from outside the solar system. These particles vary in size and so have the potential to inflict a substantial amount of ionizing damage if they impact a spacecraft. Solar particle events send ejected protons, electrons, alpha particles, and heavier particles through the solar system and are associated with the sun's activity; such events include solar flares and coronal mass ejections.

MiRaTA will be placed into an elliptical 820 km x 450 km, 99 degree inclination orbit. Because this orbit is polar, it will be subject to the radiation concentrated around the poles. Assuming 1 mm aluminum average shielding around the spacecraft, the expected total dose over a 1-year mission for MiRaTA is 9.36 krad [35]. To verify that the MiRaTA spacecraft would be able to survive this total dose, various electrical components were exposed to doses at or above the expected 9.36 krad value to determine their points of failure [36].

The device used for radiation testing was the Gammacell 220 (Figure 2-11), a cylindrical Cobalt-60 radiation chamber which releases gamma rays at a predictable and measurable rate. A number of integrated circuits and SD cards were placed into the chamber and were commanded to operate during radiation exposure. Three different dosage levels were used: 8 krad, 16 krad, and 24 krad [35]. The components were inspected before and after radiation exposure to yield pre- and post-exposure performance characteristics. Of all components tested, only one failed, and this occurred at the 24 krad level. This component is the ADG452, a monolithic, single-pole, single-throw switch and was tested for the following characteristics: rise time, prop. rise time, and fall time [35]. Following the exposure to the 24 krad total doseage level, these performance characteristics fell outside of their normal, pre-exposure, operating ranges [35]. Fortunately, however, every component that was tested passed on the 16 krad level, which is greater than the 9.36 krad expected total dose with substantial margin, which gives confidence that the avionics hardware will not have issues on-orbit for MiRaTA due to radiation exposure.

Figure 2-11: Gammacell 220 Radiation Chamber

2.4.2.3 Vibration Testing

Vibration testing on satellite programs is used to determine the spacecraft's resonant modes, validate the integrity of the structural design, and verify that the spacecraft will survive the vibrations to which it will be subject during launch. Vibration testing on MiRaTA will take place during three different testing regimes. The first will utilize the mass mockup, a full-scale structurally representative model of the spacecraft with representative masses, center of gravity values, and moments of inertia for all components and the spacecraft as a whole. The second and third will use the MiRaTA flight model.

As with all environmental testing, the LISP dictates the expected parameters of the vibration test – in this case, the vibration levels, and number of vibration iterations in each axis, which must be met. The vibration sequence on the MiRaTA project will be as follows:

- Y-Axis
    - Pre-sine sweep
    - Random – lift-off vibration profile
    - Random – transonic vibration profile
    - Post-sine sweep
- X-Axis
    - Pre-sine sweep
    - Random vibration profile
    - Post-sine sweep
- Z-Axis
    - Pre-sine sweep
    - Random vibration profile
    - Post-sine sweep
- Tune radio to beacon transmit frequency
- Wait >45 minutes and listen for beacon

Each vibration profile has three corresponding response levels dictated by the LISP (in ascending order of response magnitudes): acceptance, protoflight, and qualification. Based on the phase in the program in which the testing is conducted, and the requirements of the LISP, teams will test to one of these levels during a given test. For the first two vibration tests conducted on the MiRaTA project, the spacecraft is tested to the protoflight levels. These two vibration tests are conducted at the MIT Lincoln Laboratory test facilities, the first time on the mass mockup and the second time on the integrated spacecraft. The final vibration test is conducted at the LISP's facilities after spacecraft delivery. This test is conducted at the acceptance level on the CubeSat integrated into the P-POD with the Remove Before Flight Pin (RBFP) removed. Pre and post-testing vibration sine sweeps are conducted on the CubeSat to ensure that nothing changed substantially or broke during testing, while the RBFP is removed to ensure that the power-inhibiting separation switches prevent the spacecraft from turning on during the vibration conditions of launch.

2.4.2.4 Shock Testing

Shock testing on a satellite program is used to simulate the shock envelope to which the satellite will be subject for a given launch vehicle. This test can be conducted in a number of ways, though it usually involves some form of pyrotechnic impact device which excites one spacecraft axis at a time, or all at once.

Shock testing will be conducted twice on the MiRaTA program. The first test iteration will be conducted on the mass mockup with a few payload and bus avionics hardware components, whose susceptibility to shock are of particular concern since they contain oscillators that may be sensitive. The second test iteration will be conducted on the integrated flight model space vehicle. Both shock tests will be conducted in a P-POD simulator, a “TestPOD,” provided by the LISP and will be tested to the LISP’s designated protoflight shock levels two times (both directions) per spacecraft axis. For the MiRaTA program, the shock test is performed using a shock fixture, a test pod, and the test article bolted onto a shock table. A nail gun is used in order to create the shock and the resulting velocities of the attached accelerometers are recorded using a data acquisition (DAQ) system. Initially, a calibration test is conducted on a representative “dummy mass” in order to ensure the correct response levels are achieved during testing (Figure 2-12).

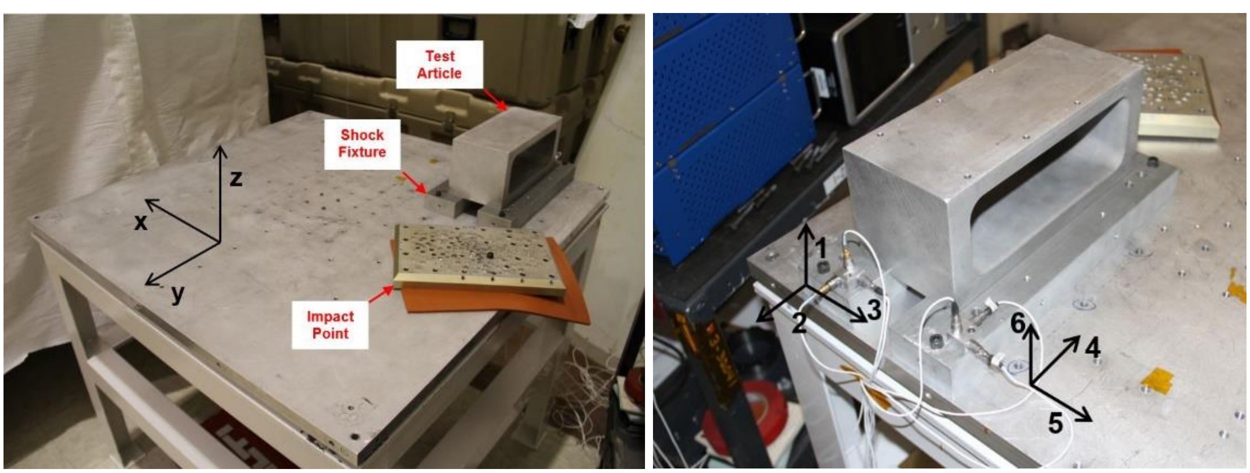


Figure 2-12: MiRaTA Shock Calibration Test Setup

At the time of this writing, only this initial calibration of the shock table with the "dummy mass" had been completed, while preparations for the actual tests were being made. Calibration test results for the accelerometer channels in the vertical (z-axis direction) are given in Figure 2-13. The solid red line represents the target protoflight response level and the dotted lines correspond to the upper and lower margin bounds on that level dictated by the LISP.

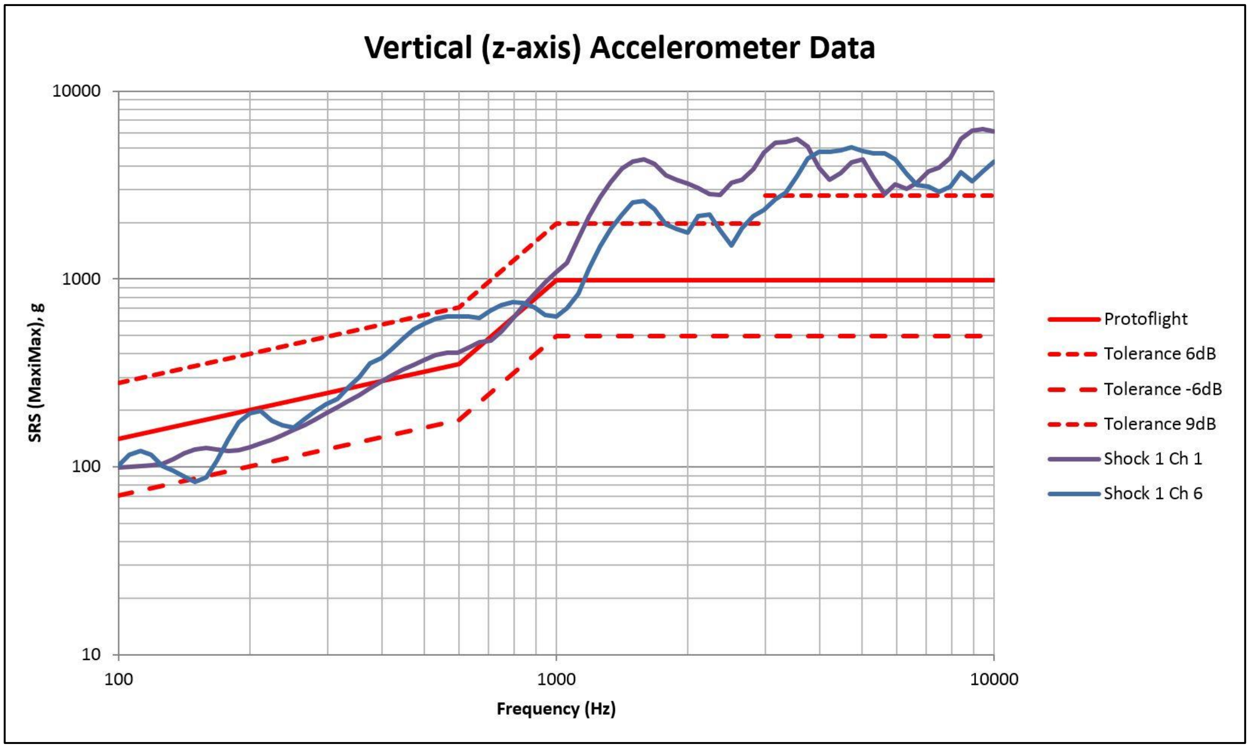


Figure 2-13: MiRaTA Shock Calibration Test Results

This data shows that the response exceeded the upper acceleration bounds in the high frequency end of the spectrum. After a number of test iterations and calibration attempts, it was determined that this would be necessary in order to get the response in the low frequencies to be within its lower bounds. Because of the nature of testing, the launch provider needs to ensure that the spacecraft will not break during launch and potentially cause damage to other spacecraft or the launch vehicle itself. For this reason, meeting the lower bounds at every point along the spectrum, even if it means over-testing at some points, was deemed acceptable, and the team continued with preparations for post-calibration testing.

Chapter 3: CubeSat Failures

3.1 Analyzing the History and Causes of CubeSat Failures

In addition to validating requirements and collecting pre-operational data, testing on a satellite is conducted for the purpose of preventing failures. Satellite failures are not uncommon. Of 6,854 satellites launched between 1957 and 2009, 732 (11%) were reported as failures [37]. In this case, a failure is defined as not meeting any of the mission objectives. Testing on a satellite program aims to mitigate the risk of such failures occurring.

Since one main focus of this research is on CubeSats, it is relevant to analyze how failures happen on CubeSat programs. CubeSats overall currently have a low success rate, with only one-fifth of missions to-date having achieved all of their mission objectives (Figure 3-1) [1] [2]. This figure shows a distribution of CubeSat mission statuses by year. "Mission status" is defined as the farthest the satellite made it into ops prior to either a planned End of Mission or mission failure. This research uses data from the St. Louis University CubeSat Database and Gunter's Space Page [1] [2]:

1) Launch failure: the CubeSat failed to make it into orbit.
2) Dead on Arrival: the CubeSat was successfully deployed from the launch vehicle and then failed.
3) Early failure: the spacecraft had at least one uplink and one downlink prior to failure.
4) Some Operations: the spacecraft is taking actions that achieve primary mission success (i.e., receiving commands, downlinking mission data), though has not achieved all mission objectives.
5) Mission Success: all primary mission objectives have been met.
?) Unknown: mission status data is not currently available.

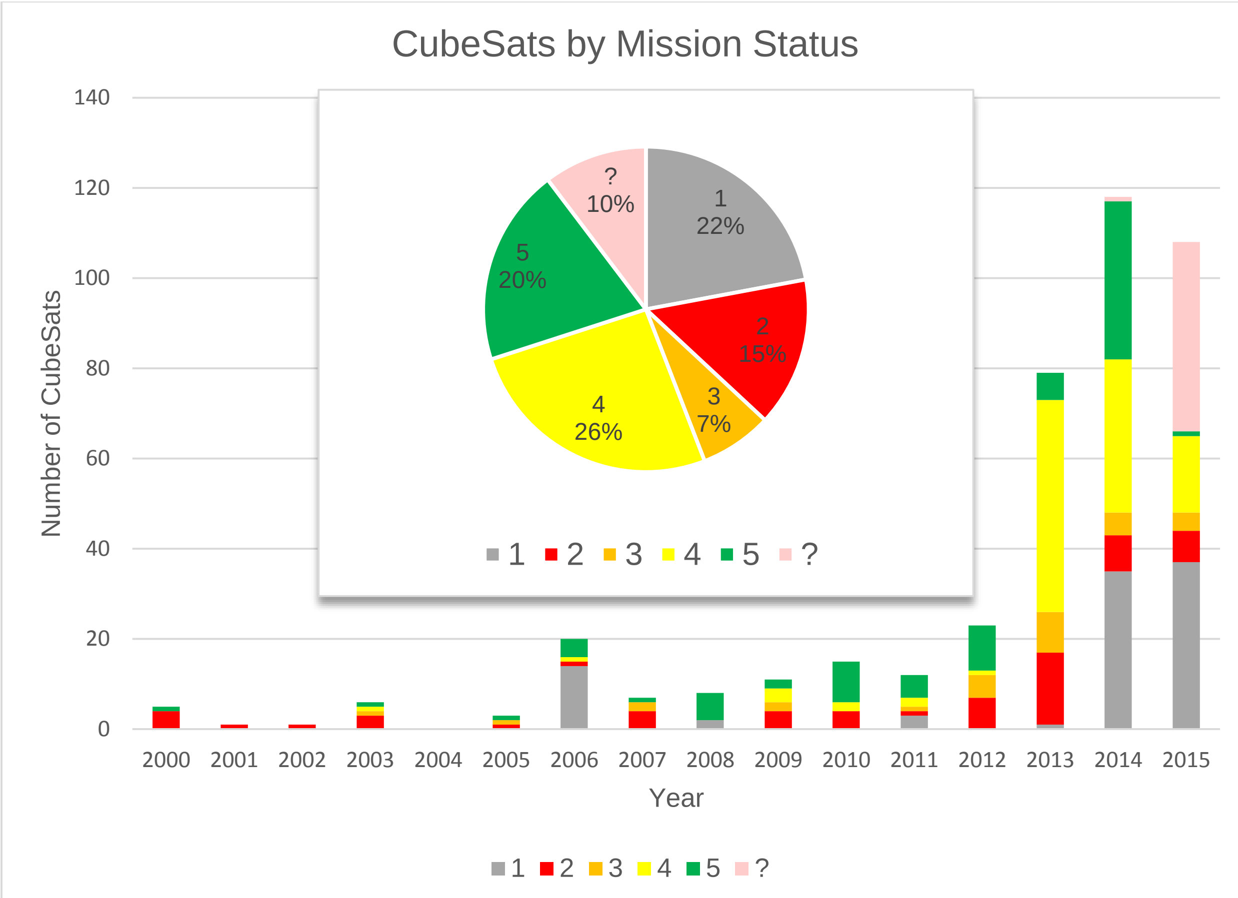


Figure 3-1: CubeSats by Mission Status [1] [2]

As compared to the satellite industry as a whole, there is undeniable evidence that CubeSats have a higher rate of failure. Over 28% of successfully launched CubeSat missions have failed to achieve any of their primary mission objectives (categories 2-3) – a stark contrast to the 11% for satellite missions as a whole [1] [2] [37]. A number of studies have been conducted on the reliability of spacecraft based on their size category ([38] [39]) which conclude that the failure rate of small satellites (0-500 kg range) is roughly twice as high as that in the medium and large size categories. One phenomenon closely linked to these failures for satellites in the small satellite category is infant mortality: when a spacecraft fails within the first couple of months after orbit insertion. Statistical analysis shows substantially higher infant mortality for satellites within the small size category versus those in the medium and large size categories (two times the decrease in reliability after six months) [38]. This raises the question: why do smaller satellite missions, such as CubeSats, fail more frequently? The remainder of this section conducts an analysis to answer this question.

Delving into the sources of failures for CubeSat missions, a common theme arises: the problem is not of a lack of testing, but rather of misplaced testing priorities [38] [39] [40]. As a result, CubeSat projects tend to conduct insufficient integrated testing. One possible reason for this is the emphasis that CubeSat groups (especially university groups) place on component testing, rather than on integrated testing. From the author's experience of working on three university satellite projects, there seems to be a misconception that if each individual component works, then the integrated system will therefore work. Components are tested as they arrive through the program (usually workmanship inspection and functionality testing). Also, university groups tend to have very tight schedules, with only one or two years for the expected turnover time from program start to delivery; without the up-front emphasis on integrated testing, combined with any delays on the program which pushes the schedule to the right, the amount of testing which is conducted on the integrated space vehicle is usually very limited. Sometimes, there is only enough time to conduct just the required amount of environmental testing and then deliver to the launch provider, lest the launch opportunity be lost. Figure 3-2 below shows the distribution of causes of CubeSat mission failures.

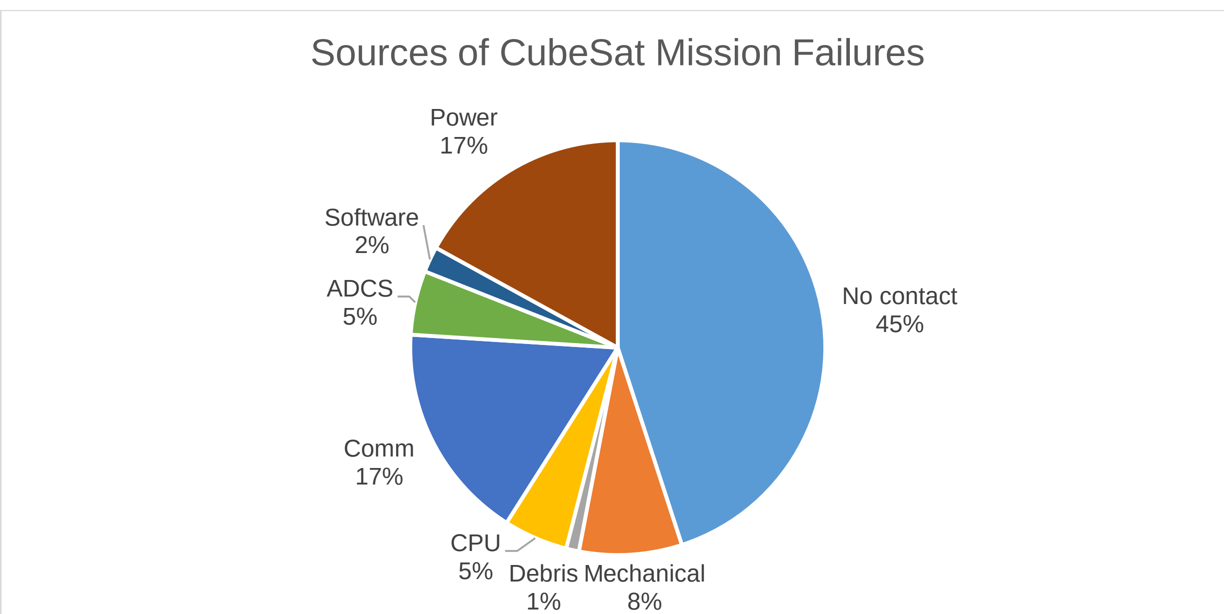


Figure 3-2: Sources of CubeSat Mission Failures [40]

Nearly half of all failures on CubeSat missions are attributed to the inability to contact the spacecraft. Analysis of mission reports shows that these failures are most often caused by a configuration or interface failure between a combination of the communications hardware, the power subsystem, and the avionics hardware (flight processor) [40]. This is a direct result of not conducting adequate testing in the flight configuration; tests to verify proper operational functionality of the space vehicle would detect such failures.

There exist additional aspects of CubeSats which contribute to their greater failure rates versus the satellite community as a whole, namely parts selection and attributes associated with their smaller size. CubeSat projects often rely on COTS components for their spacecraft build, and this is done for the following reasons: COTS components are typically cheaper than their industry-level space rated technology counterparts; vendors are now producing and selling COTS assemblies (e.g. avionics stacks, batteries, EPS, ADCS units) which decreases time CubeSat groups have to spend on custom development; with the continued miniaturization of technologies, as CubeSat complexity increases, the technical capabilities of student engineers can be exceeded, and already-built COTS assemblies can mitigate this issue. COTS components do not come without some disadvantages, however. First, COTS components achieve their competitively cheaper price as compared to their industry-level space rated technology counterparts, through decreased production and testing times, in addition to the use of lower-quality hardware. This inherently adds increased risk to COTS hardware. Typically COTS components undergo rapid, mass production, as opposed to more meticulous, single-build production for more expensive custom space-rated hardware. While the more complex COTS components, especially assemblies, undergo relatively more testing than other COTS hardware, it is still substantially less than that of standard space-rated hardware, and frequently post-testing modifications and reworks are conducted (and are deemed acceptable) on flight COTS hardware.

Quite possibly the greatest risk factor to using COTS equipment, however, is not the attributes associated with the hardware itself, rather, it is with the effect it has on testing and integration within the satellite projects that use them. Integration and test strategies for spacecraft revolve around component delivery dates: when certain types of testing can begin, to include hardware interface and assembly testing, depends entirely on when the hardware is delivered to the team. This means that delays in component deliveries can prove extremely detrimental, even fatal, to satellite projects. When CubeSat teams outsource to commercial vendors for COTS components, they must understand that the greater the percentage of the spacecraft which is allocated to COTS components, the greater the abdication of control of the satellite's schedule to groups outside of the satellite team. Delays in COTS component deliveries are by no means an uncommon occurrence either; of the six COTS component assemblies purchased on the MiRaTA project, for example, only two arrived on or before the agreed-upon delivery date, and two of the four which were late arrived six months or later after the quoted delivery date. This had huge ramifications on the project: the testing and integration schedule was modified many times, and began to depend entirely upon the dates which the components would be delivered; some larger assembly-level testing which was planned months prior had to be eliminated due to time constraints caused by the late deliveries; last but not least, team member and management time was wasted on increased communication between the team and vendors regarding hardware after the missed delivery dates.

The last aspect of CubeSat projects which contributes to their increased failure rates is their smaller size. Due to the nature of their deployment as auxiliary or secondary payloads on launch vehicles, CubeSats face some of the tightest size and weight constraints of all satellite projects. Though the exact requirements depend on the specific deployer (P-POD, NanoRacks, ISIPOD, etc.) being used, this typically equates to a size of 10 cm x 10 cm x 10 cm and less than 1.33 kg per CubeSat "U," with a maximum of 3U for the majority of deployers today. Because of this, there usually is not enough remaining space or weight to integrate redundant systems in CubeSats. As a result, CubeSats often have many more "single-string" subsystems – systems where a single anomaly could result in the failure of the entire spacecraft. In addition, the components which are integrated into CubeSats are fitted very tightly together to make sure everything fits, which can pose significant thermal issues across the spacecraft with components being exposed to greater temperatures than they otherwise would have been [38].

3.2 Case Study Analysis: MicroMAS

For the purpose of this research, it is helpful to analyze the MicroMAS mission as a case study for a CubeSat with a mission failure. A description of post-deployment operations is discussed, with particular emphasis placed on the methods by which the team tracked down the likely source of the failure, in order to provide a framework for how other CubeSat teams can approach investigating failure modes that may emerge on their spacecraft. This case study should be of particular interest to CubeSat engineering groups as the type of the failure MicroMAS experienced, a failure to establish repeated successful contacts with the spacecraft, is the most common type of failure to date for CubeSat missions [40].

MicroMAS was launched on an Antares 120 rocket on July 13th, 2014 as part of the Orb-2 ISS resupply mission. From this point its deployment from the ISS NanoRacks CubeSat Deployer was delayed until March 4th, 2015. Shortly after it was finally released from the ISS, ground contact was made.

Unfortunately for the MicroMAS mission, ground contact was only established two more times after the initial contact on March 4th, 2015. In total, about 20 minutes of usable data was downlinked, which mostly consisted of telemetry information for the ADCS, power, and avionics subsystems. Ground controllers were never able to command the spacecraft to enter its payload operational mode and, as such, no data from the microwave radiometer was collected.

Though MicroMAS was not able to accomplish its primary mission objectives related to its radiometer payload, the telemetry data which was collected did provide useful information about the spacecraft, answering many important questions:

- After the delayed deployment, was the battery still charged and healthy?
    - Yes, after deployment on March 4th the battery charge was 7.8 V (6.5 V minimum) and fully charged to 8.2 V by the March 5th contact.
- Did all the solar panels successfully deploy?

    - At least three out of four did, which also confirms proper functionality of the battery and EPS subsystems, as the voltage lines out of these systems needed to function properly to be able to fire the TKDs. The analysis is still underway to examine the end state of the remaining panel.
- What was the thermal environment like?
    - It was cold at deployment, then varied with orbit and beta angle. The batteries (which are the most sensitive) had a temperature of -8 deg C March 4th and 10 deg C on March 5th, well within their operational range.
- What can we tell about the spacecraft attitude?
    - The spacecraft tumbled at a rate of 10 degrees/second on March 4th, 7 degrees/second on March 5th, and 6 degrees/second on March 9th. In addition, from the telemetry, some software mapping errors were identified for the Earth Horizon Sensors and sun sensors.

These answers provided useful evidence for the proper functionality of many of MicroMAS' subsystems. This information also proved useful for the follow-on projects to MicroMAS which were able to modify their hardware and software based on the MicroMAS data. Finally, this information aided in tracking down the cause of the failure on MicroMAS – a critical task in order to ensure the follow-on projects would not face the same fate.

One of the key tools used to track down the source of the failure on MicroMAS was the power subsystem telemetry data. The current values across the battery and EPS voltage lines were analyzed to find indicators of the communication subsystem transmit attempts, which, if successful, would show moments of peak current draws during transmits. Figure 3-3 and Figure 3-4 below show plots of the current telemetry data for MicroMAS during two separate transmit attempts. Figure 3-3 shows the current draw response during a successful transmit attempt: the raw bus voltage line, which powers the spacecraft radio, has a current draw spike during each of its successful transmits. Also pictured on this plot are the 5 V and 3.3 V voltage lines, which remain near zero as the components which they power are off during this transmit. Figure 3-4 shows the telemetry data for a failed transmit attempt. During the time period over which this plot spans, the spacecraft radio was commanded to transmit; the corresponding spike in current draw on the raw bus voltage line indicative of a successful transmit was never seen, leading to the conclusion that the transmit had failed. The transmit depicted in Figure 3-3 was the last successful attempt for the MicroMAS mission.

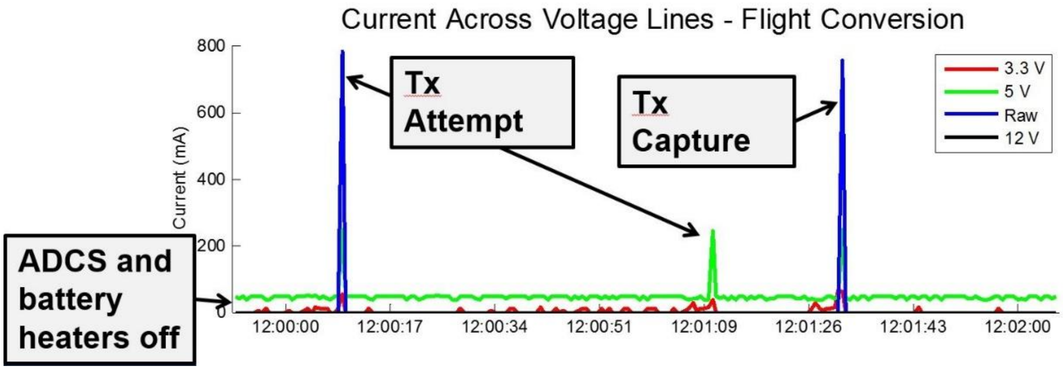


Figure 3-3: MicroMAS Current Telemetry Data – Successful Comm Attempt

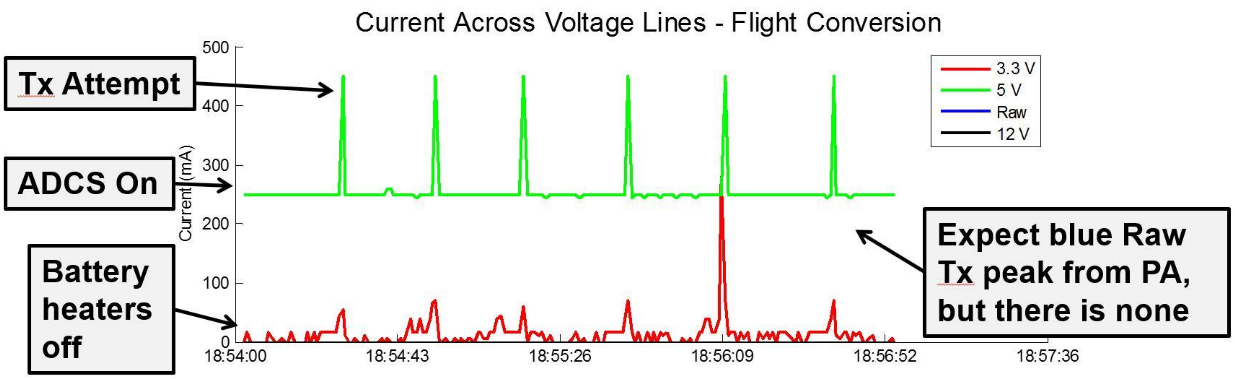


Figure 3-4: MicroMAS Current Telemetry Data – Failed Comm Attempt

Once it was determined that the source of the failure was a communications fault, the next step was to determine what specifically within this area failed. A communications fault leaves open the possibility for a wide range of source failures, from a failure of the ground station to a problem with the spacecraft radio itself, so a "fault tree" was created and then possible sources were analyzed one by one to narrow-in on the root cause (Figure 3-5). By conducting this analysis, the basic communications operations and ground station were fully verified: they were not the root cause of the failure. All potential space vehicle-related root causes were fully verified as well, with the exception of the potential for a power loss to the radio. There was insufficient data to fully verify this was not the root cause of the failure, though from the data which was present (especially the data which showed the successful deployment of three of the four solar panels, which required successful operation of the power lines out of the EPS), it was concluded that this was an unlikely root cause. Finally, aspects of the spacecraft radio itself were analyzed. Many of these aspects were determined to have worked properly, which then left the only remaining possibility for the root cause of the failure to have been a malfunction of the radio hardware.

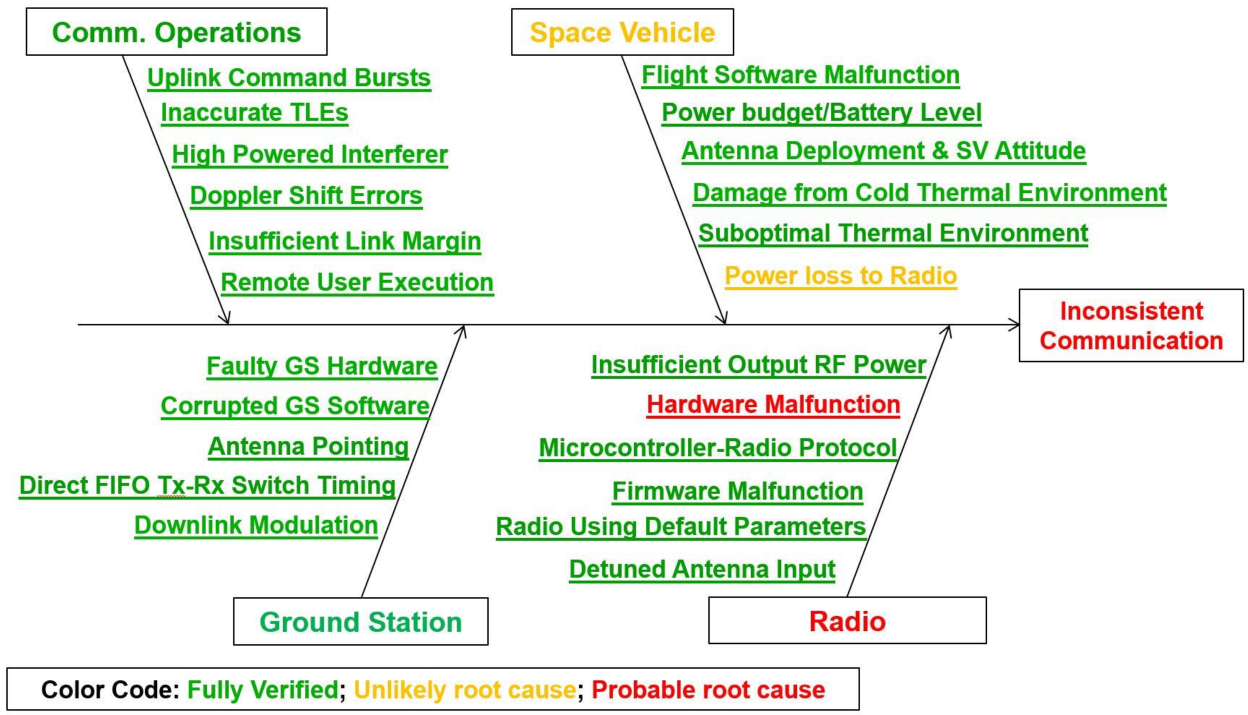


Figure 3-5: MicroMAS Communications Fault Tree

With the most probable root cause of MicroMAS' inconsistent communication failure having been determined to be radio hardware malfunction, the final step in this process was to determine what specific piece of radio hardware malfunctioned. This was of particular importance to the MicroMAS team as the follow-on CubeSat mission, MiRaTA, was slated to use a nearly identical radio. Figure 3-6 depicts the results of this analysis. In the end, it was determined that the most likely root cause of the failure was in the radio transmit chain. At one point during the testing and integration phase of the program, components within the radio transmit chain were damaged when the radio was powered on and transmitted unterminated (uncommanded upon power-up). The most obvious damage was to the power amplifier, which was replaced. Other components within the transmit chain were inspected and tested but were not replaced.

Later, during integrated thermal vacuum testing, the radio experienced at least a couple transmit anomalies at cold and hot thermal vacuum temperatures. Due to previous schedule slips which delayed this thermal vacuum testing until just before the spacecraft delivery, there was insufficient time to adequately examine these anomalies. In retrospect, after the transmit chain was damaged during testing, it probably would have been wiser to replace the entire radio board rather than try to go component-by-component in search of what was damaged and what was not. The other components which were inspected and determined to still function properly very possibly could have retained a latent failure which was not detected during inspection, only to resurface during integrated operations. While the investigation of the MicroMAS failure is still underway, with other factors such as one of the four solar panels failing to deploy (potentially restricting the antenna deployment) still in question, the failure of the transmit chain within the radio remains at least a contributing factor, if not the most probable root cause, of the inconsistent communication [41].

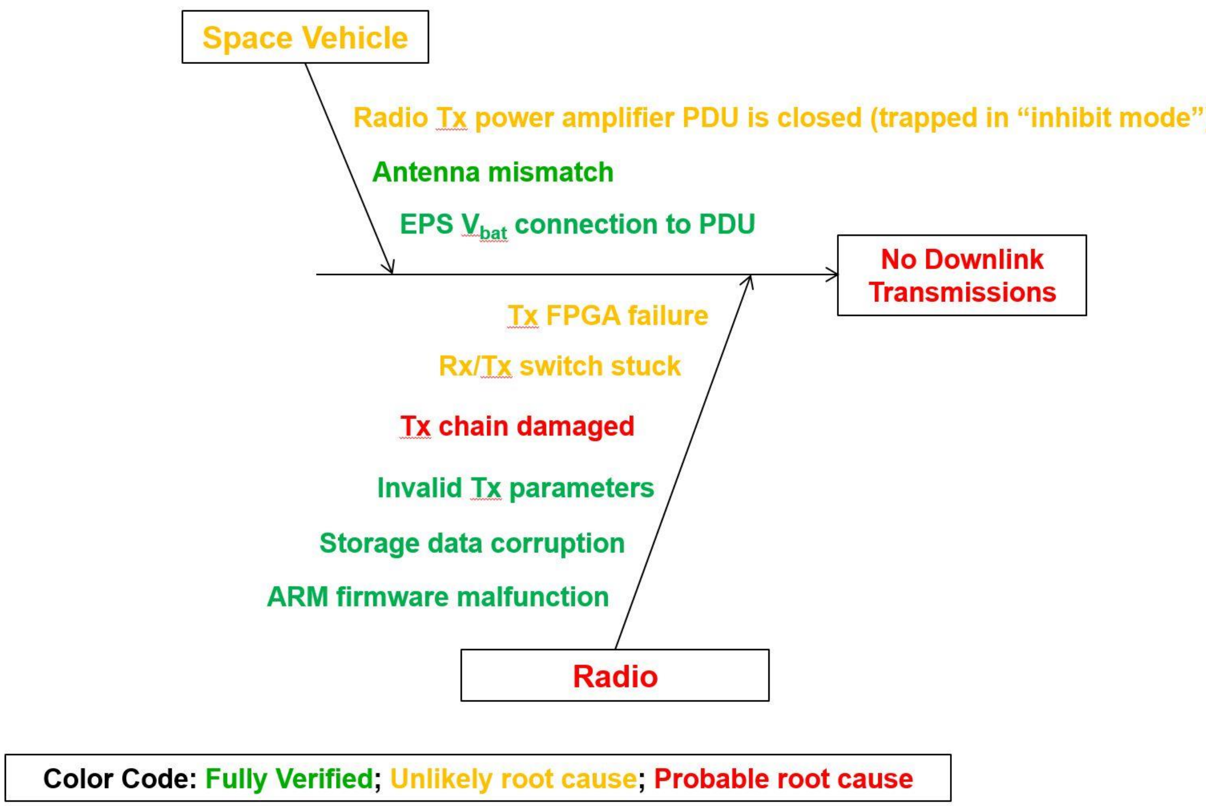


Figure 3-6: MicroMAS Radio Hardware Fault Tree

The lessons learned from the MicroMAS mission and its post-failure analysis proved to be extremely useful for its follow-on missions. These lessons learned are listed here in the event that they may aid other CubeSat engineering groups:

- An aggressive risk posture (as is associated with CubeSat projects) necessarily constrains analysis – as such, an efficient and well-planned test program is essential.
- With the split-nature of the MicroMAS team (professional engineers building the payload at one location and graduate students building the bus at another), there should have been earlier engagement of the experienced staff to support bus subsystem design and testing. A team-wide common flight software and documentation repositories (with sharing and version control) is needed.
- MicroMAS' tight schedule did not permit necessary regression testing on repaired components. This, when coupled with a tight budget which did not support the purchase of flight spares, proved to be fatal for the project. Flight spares are critical.
- CubeSat projects need a greater emphasis on "test-as-you-fly": testing in the fully integrated state. This was precluded by the budget and schedule for MicroMAS – this needs to be prioritized and planned for early in a program.

3.3 Learning from Failures: Second Iteration CubeSat Missions

Experience in satellite engineering matters, as the lessons-learned from the first satellite are carried over to the second. Across the board, CubeSat missions are much more successful on a group's second attempt (Figure 3-7). The below figure shows two pie charts: the left chart is a mission status distribution of all successfully launched CubeSats which were a group's first; the second chart is a mission status distribution of all successfully launched CubeSats which were a group's second successfully launched mission.

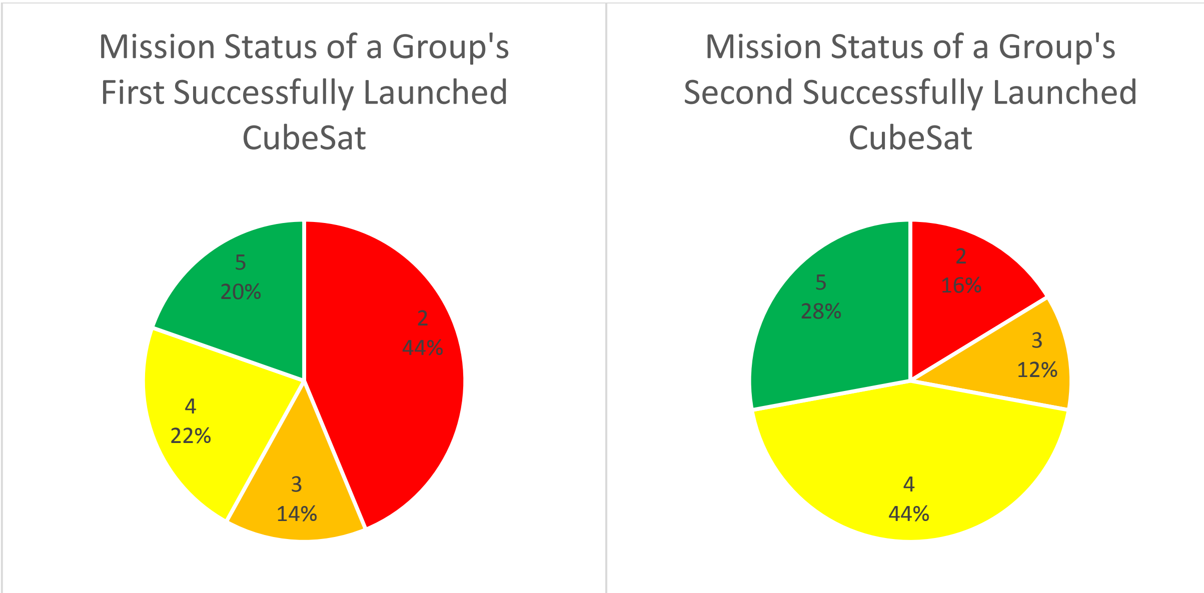


Figure 3-7: CubeSat Mission Improvements from First to Second Attempt [1] [2] [42]

As can be seen in the above figure, the success rates of CubeSats which are built by groups which already have the experience of a prior CubeSat are much higher. The percentage of CubeSats which are able to successfully conduct primary mission operations (categories 4 and 5) increase from 42% on a first CubeSat to 72% on a second (30% increase)! The percentage of CubeSats which fail on-orbit, consequently, decrease by this same amount.

Looking at university groups specifically, the numbers are even worse. 50% of all successfully launched university-built CubeSats, which were that group's first, fail to establish successful communication with the ground (they are "dead on arrival" – mission status: 2) [2]. This number decreases substantially on a university group's second CubeSat, however, to 17% [2]. These statistics offer credence to an unsurprising fact: experience contributes to a satellite's success. Even if individual team members are inexperienced (university groups which successfully launch a second CubeSat normally do so years later, at which point some or all of the previous student engineers would have graduated), organization experience contributes immensely to mission success. This is likely due to the fact that lessons on what could have been done better on the previous satellite are carried over into successor missions (and can even be used from the start of a mission in the design of that mission's schedule and integration and testing strategy).

Considering the fact that experience contributes to mission success, why not leverage that experience by artificially introducing it to a group? Rather than building a first satellite from scratch, start-to-finish, and then launching it with the hopes that everything works, a much more prudent strategy for an inexperienced group would be to build a "beta version" of their satellite first, so that they can adequately test it in the integrated state, discovering modes of failure that they otherwise would not have found until their satellite made it to orbit previously.

There are certainly points to be made against this strategy. The first is cost: clearly program cost will increase if a group is building two satellites rather than just one. The second is time: it will take longer to build and test two satellites as well. The counterarguments to these points reveal the value of this "beta build" strategy: first, consider the fact that only one-fifth of all first-attempt CubeSats have accomplished their mission objectives. Considering this, if the groups which fail still have the intent to accomplish their objectives, they must build a second satellite and try again. As will be shown in sections 4.1 – 4.3, this path (building a second satellite after an unintended failure) is much more expensive than if the group went into the project with the strategy to build two satellites from the beginning. This cost difference is increased further if the team saves time by building the satellites in tandem and varies certain other parameters such as testing and payload integration, which will be discussed in chapter 5.

It is true that it will always be less expensive to have built only one satellite if that first satellite were completely successful (though again, it is emphasized that this has not been the case for 80% of CubeSat missions thus far). In this field, however, which is so intolerant to operational faults, errors, and failures, why take this extremely risky posture when, for a slight compromise in cost and schedule, it can be drastically mitigated? The remainder of this research discusses this in-depth.

Chapter 4: Total Program Cost Analysis

4.1 Analysis Overview

To understand the impact that different build strategies have on CubeSat programs, the total program cost must be analyzed. Note that the "total program cost" being analyzed here is broader than a program's budget. A program's budget is typically what is tracked by a program manager or systems engineer; it is the "cost-to-build" value for the satellite (or set of satellites). Total program cost, on the other hand, is defined here to include not only the budget, but also additional costs such as external operator costs during the mission, launch cost, and the cost to pay the members of the satellite team. The total program cost can be thought of as the total amount of money spent taking a satellite program from its conception to mission completion and is broken down into the following categories:

- Staff Cost – the cost to pay all personnel on the program. This includes management, engineering staff, and additional technical staff. For a university program, management can also be faculty; engineering staff, upper-level students (graduate, if applicable); and additional technical staff, lower-level students (undergraduate, if applicable).
- Hardware Cost – the cost of all engineering and flight components, as well as ground support equipment. This includes the cost of repairs, reworks, and replacements.
- Environmental Testing Cost – the cost of environmental testing on the program. This includes thermal, vacuum, radiation, vibration, and shock testing (discussed in Section 2.4.2). The cost of testing is defined in terms of personnel hours dedicated to preparing for, running, and analyzing the results of environmental tests.
- Launch Cost – the cost to launch each satellite (includes launch integration services costs).
- External Operator Cost – the cost of external operators (those individuals in separate ground stations hired to aid with operational tasks such as tracking, commanding, site maintenance, fault debugging, and data analysis).

The 3U form factor is used throughout the total program cost analysis. This is because it makes up the majority of the CubeSat industry (at 53%, with the next closest being 1U CubeSats at 31%), as well as for the practical reason that the author's experience on the MicroMAS, MicroMAS-2, and MiRaTA 3U CubeSat projects can be leveraged in the analysis. This analysis is less focused on the cost values themselves, than on the comparative differences between a baseline program and those which experience failures or use alternate build strategies. Using a common 3U form factor is also necessary in order to make comparisons and identify these relative differences, even if absolute values may not be highly accurate. The following build strategies are compared, where λ corresponds to the attempt:success ratio, "U" corresponds to an unplanned failure on the first spacecraft, and "P" corresponds to an alternate build strategy, in which the team planned to build two flight spacecraft from the beginning:

- 1:1 λ: CubeSat baseline (3U size). One spacecraft is built and is a total success.
- 2:1 λ U: The baseline strategy is used, but the first spacecraft fails. The team then rebuilds and launches a second spacecraft from scratch, but in shorter time because of experience.
- 2:1 λ P 1: Beta build strategy 1, in which the team plans to build two spacecraft from the beginning. The time after the first spacecraft launch is shortened because both spacecraft are built and tested simultaneously.
- 2:1 λ P 1: Beta build strategy 2, which maintains the same parameters from 2:1 λ P 1, but in this case the team does not integrate the expensive payload on the first spacecraft.
- 2:1 λ P 1: Beta build strategy 3, in which the team does not launch the first spacecraft at all. Instead, they lengthen testing on the ground for both spacecraft. This still assumes the first spacecraft is completely expendable.

The analysis is organized as follows: first, the baseline total program cost for a 3U CubeSat program is determined. This baseline assumes that (1) the CubeSat team builds one completely integrated CubeSat, and that (2) the mission is a complete success, with the CubeSat meeting all of its mission objectives. The total program cost is then reanalyzed for a program which fails on its first attempt and then decides to rebuild the satellite for a second attempt. It is assumed for the analysis of this program that the second satellite is successful; as such, this program is designated as "2:1 λ U (Unplanned Failure First Iteration)," where λ is the attempt:success ratio. The ideal attempt:success ratio is 1:1, where the satellite is successful (achieves its mission goals) on its first attempt, which is represented by the baseline. Chapter 5 then conducts cost analyses for projects which utilize various versions of the CubeSat "beta build" strategy, in which the program begins with the plan to build two satellites. Because the beta satellite is another complete, integrated, version of the satellite, this program can also be thought of as having a 2:1 λ, though it is planned (and is given the "P" designation accordingly). Testing and build parameters are varied across three suggested beta build strategies, with the attempt to decrease cost while maintaining satellite reliability. The total program cost of these beta build strategies, as well as their relative merits and probability for success, are then compared to those of the first two programs. The goal of this analysis is to see (1) whether the "beta build" strategies cost less than a program with an unplanned 2:1 λ and (2) if these strategies present a potentially worthwhile trade in cost for decreased risk as compared with the traditional 1:1 λ baseline strategy.

Information from the MiRaTA, MicroMAS, and MicroMAS-2 projects such as hardware cost, staffing, program duration and time between milestones, and testing are used to construct this cost analysis. Note that due to export control requirements and non-disclosure agreements (NDA) between the CubeSat team and vendors, all cost values used throughout this analysis are not exact, rather they are representative "ballpark values." In addition, cost values are represented holistically (the cost of a specific piece of hardware, for example, is not called out). This analysis shows the distribution of spending on a CubeSat program, when the spending occurs, and the resulting end cost of the program, for which these representative cost values are sufficient.

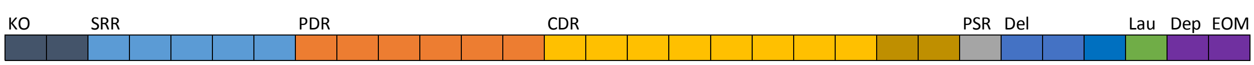
4.2 Baseline – 1:1 λ

The general theme of this cost analysis is to break down the total amount of money spent per day within each spending category, and then to sum these values to find the total cost within each category. All the categories are then combined together to form the total cost of the program. It should be noted that, for the purposes of this analysis, it is not of concern who spends the money within a particular category, simply that the money is being spent.

For this first analysis, a 30-month program duration is used. This includes all time from the initial kickoff through mission completion. This duration is used because it reflects the actual program duration for the MiRaTA CubeSat (though different dates are used in this analysis), and because it is common to see a two year kickoff-to-delivery timeframe with six months post-delivery time for spacecraft of this size and complexity (during which a 60-day operation period takes place). Table 4-1 below breaks down the program duration for this mission by the length of time between each program milestone. The following abbreviations are used for certain milestones: System Requirements Review (SRR), Preliminary Design Review (PDR), Critical Design Review (CDR), and Pre-shipment Review (PSR). Note that on the graphical depiction, each block corresponds to one month, and the darker shaded portion at the end of the post-CDR time corresponds to time allocated to integrated space vehicle testing.

Table 4-1: Nanosatellite Program Duration by Milestone

Milestone	Kickoff	SRR	PDR	CDR	PSR	Delivery	Launch	Deploy	EOM
Time until next (months)	2 mo.	5 mo.	6 mo.	10 mo.	1 mo.	3 mo.	1 mo.	2 mo.	End
	Time to delivery: 24 mo.					Post-delivery time: 6 mo.			
	Total program duration: 30 mo.								



It is important for this analysis to break up the program into periods between milestones because it is between these periods that the greatest changes in program cost are seen. Hardware purchasing, as well as the amount of staffing and testing required, varies based on the stage in the program. Within each spending category, charts showing the progression of the daily spending are presented, with milestone dates marked.

4.2.1 Staff cost

Staff cost is defined as the cost to pay all personnel on the program. This includes senior staff, engineering staff, and additional technical staff. For a university program, which has faculty and students rather than employees, the faculty and additional professional engineer and/or management support may be thought of as the senior staff, upper-level students (graduate, if applicable) are the engineering staff, and lower-level students (undergraduate, if applicable) are the additional technical staff.

To analyze the staff cost, the annual cost to have each member of the team is broken down into a daily cost. While it is true that the team members will not work every single day of the year (hopefully!), for the purpose of the analysis, it suffices to spread the annual cost over 365 days – it is the trends with spending as team members are added or removed from the program, and the total cost, rather than each individual day's cost, which matter most. For this reason it does not matter if weekly cost is modeled as being spread over a five-day versus a seven-day work-week. This example project is modeled as a university program, whose senior staff can correspond to both professional engineering and/or management staff, as well as university faculty. A rough annual salary estimate for this group of individuals is used as the annual cost for senior staff. For graduate students, the cost of funding that individual (usually tuition plus a stipend) is used as their cost. Similarly, undergraduate researchers on programs are typically given a stipend for their part-time efforts, and this stipend is used as their cost. These values vary across programs and across individuals; for this analysis, an average value is assigned to each team member group to represent the different costs of different individuals on a team (Table 4-2).

Table 4-2: Nanosatellite Staff Member Annual/ Daily Cost

	Senior Staff	Upper-level Students	Lower-level Students
Annual Salary	$100,000	$50,000	$5,000
Daily Cost (Rounded)	$274	$137	$14

The staffing over the course of the program is tracked and plotted. Figure 4-1 shows the number of each type of staff member over the course of the program. Note that the staffing typically changes around the times of milestones, as work demands increase or decrease. Also, other factors affect staffing on university projects as well, such as the university's academic calendar, students' varying graduation times, and periods between semesters when additional undergraduate researchers may be utilized.

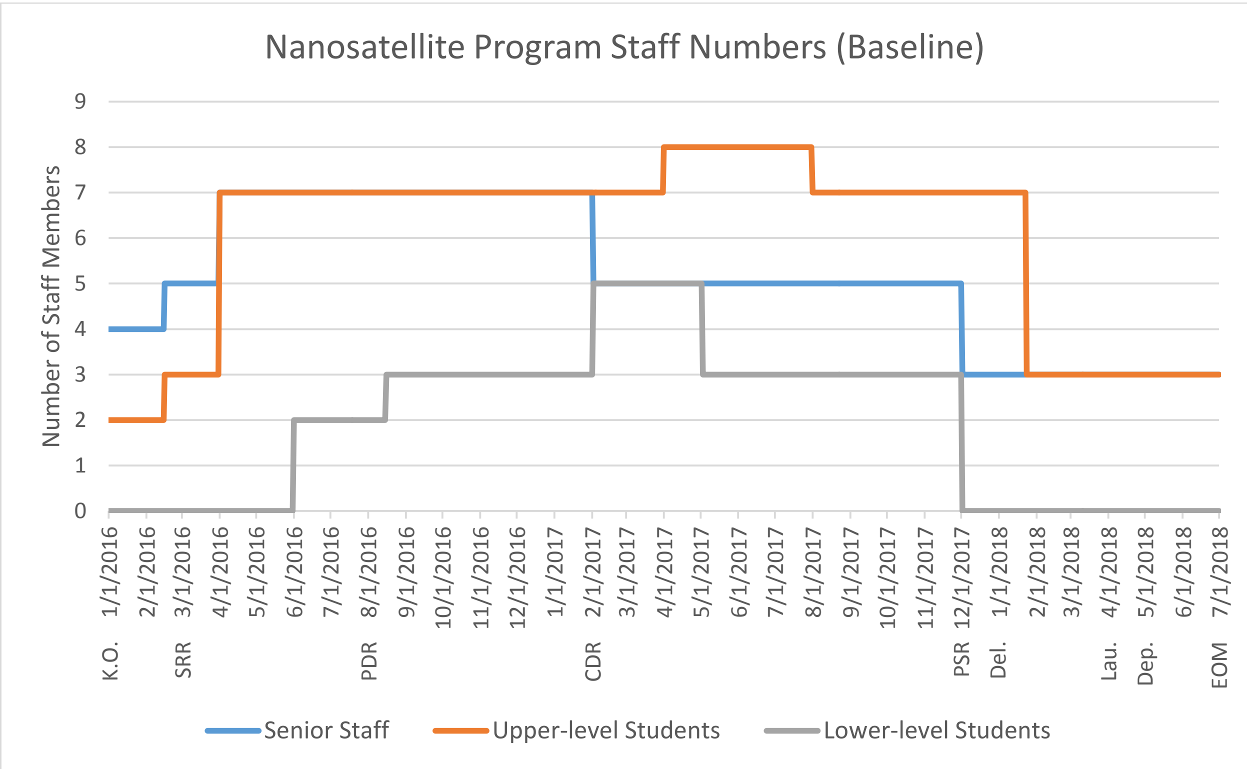


Figure 4-1: Nanosatellite Program Staff Numbers (Baseline)

The daily cost of each staff member is then multiplied by the number of staff members to yield a plot of the amount spent per day on staffing (Figure 4-2). The period of greatest staffing on this example project is between CDR and PSR, which corresponds to the period of the greatest work demands on the project. Despite this, for this particular project the senior staff numbers actually decreased during this time, as they were needed on other projects; for this reason, the period of greatest expense is actually between PDR and CDR when their numbers were greatest. The daily staffing spending is then integrated to determine the cumulative amount spent over the course of the program (Figure 4-3). The total amount spent on staffing by the end of this program is slightly over $2 million.

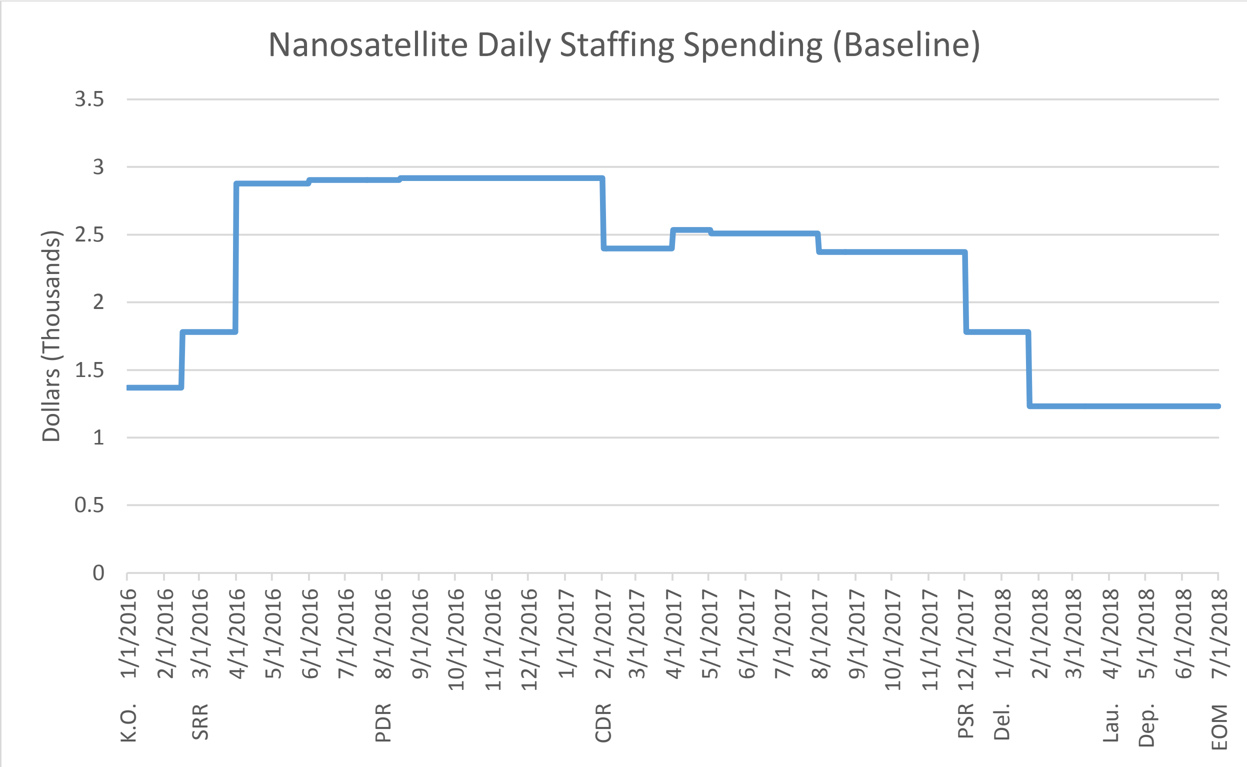


Figure 4-2: Nanosatellite Daily Staffing Spending (Baseline)

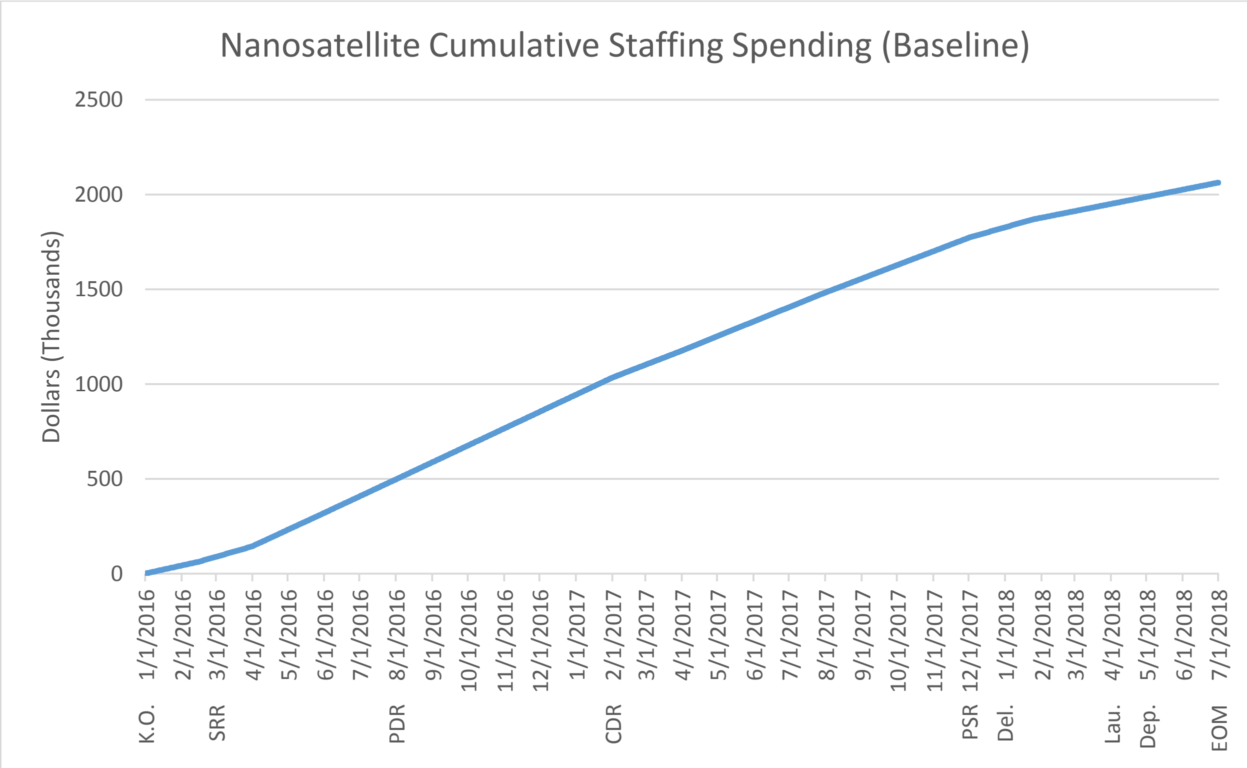


Figure 4-3: Nanosatellite Cumulative Staffing Spending (Baseline)

4.2.2 Hardware cost

Next, the cost of hardware on the program is analyzed. Hardware cost includes the cost of all engineering and flight components (and repairs, rework, or replacement, as applicable). This also includes the cost of all ground support equipment.

Similar to the program staff analysis, the hardware spending is tracked on a daily spending basis. For hardware, spending corresponds to Request for Purchases (RFP) and Purchase Orders (PO). The point at which money is actually spent for a purchase can be interpreted in one of two ways: either at the time the PO is received by the vendor, or at the time of product delivery. For this analysis, the former is used as the "money spent" definition because this is the time at which money is committed and can be considered no longer part of the available budget.

Figure 4-4 shows the daily hardware spending for MiRaTA and Figure 4-5 shows the cumulative hardware spending (due to vendor information being proprietary, which cost values correspond to what components cannot be disclosed. In addition, values are not exact, rather, representative). For this example project, everything which is not the payload is considered a part of the bus spending, including Ground Support Equipment (GSE) and the space vehicle structure. For this example program, no complete models of the payload were purchased (rather only individual components) due to its high cost and the program's limited budget.

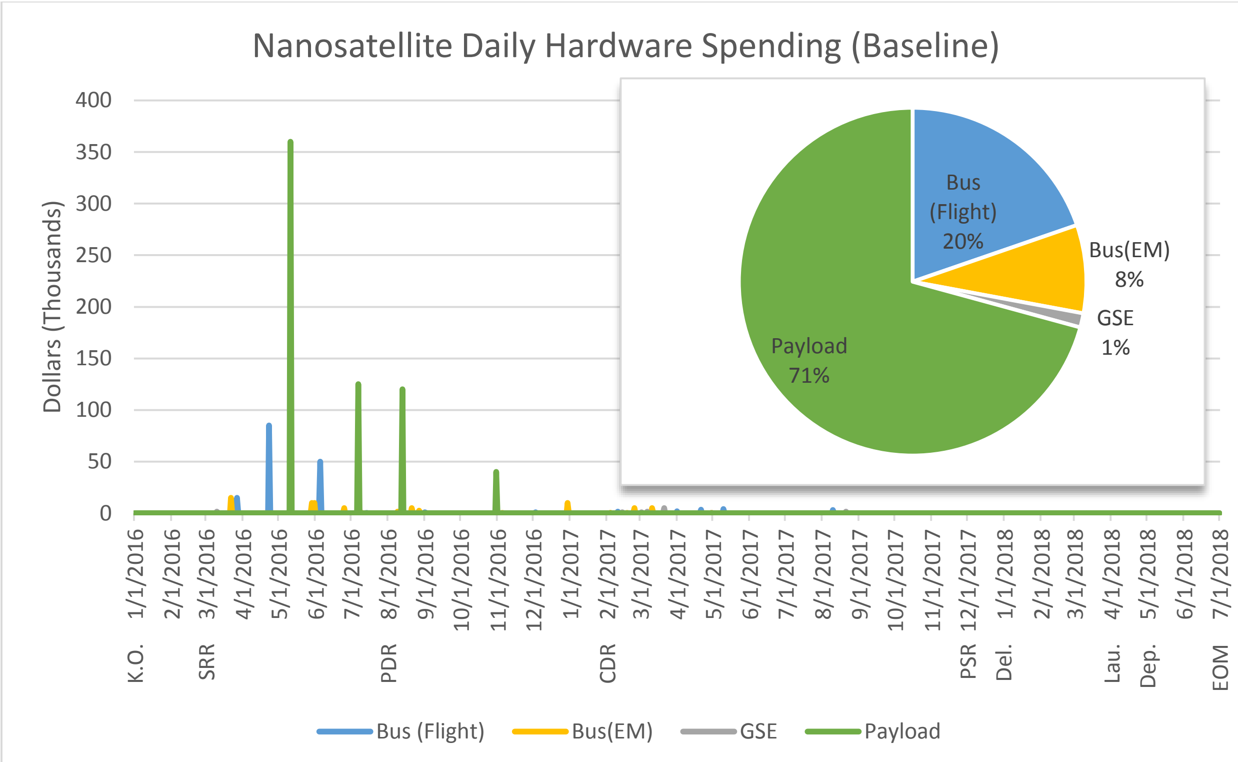


Figure 4-4: Nanosatellite Daily Hardware Spending (Baseline)

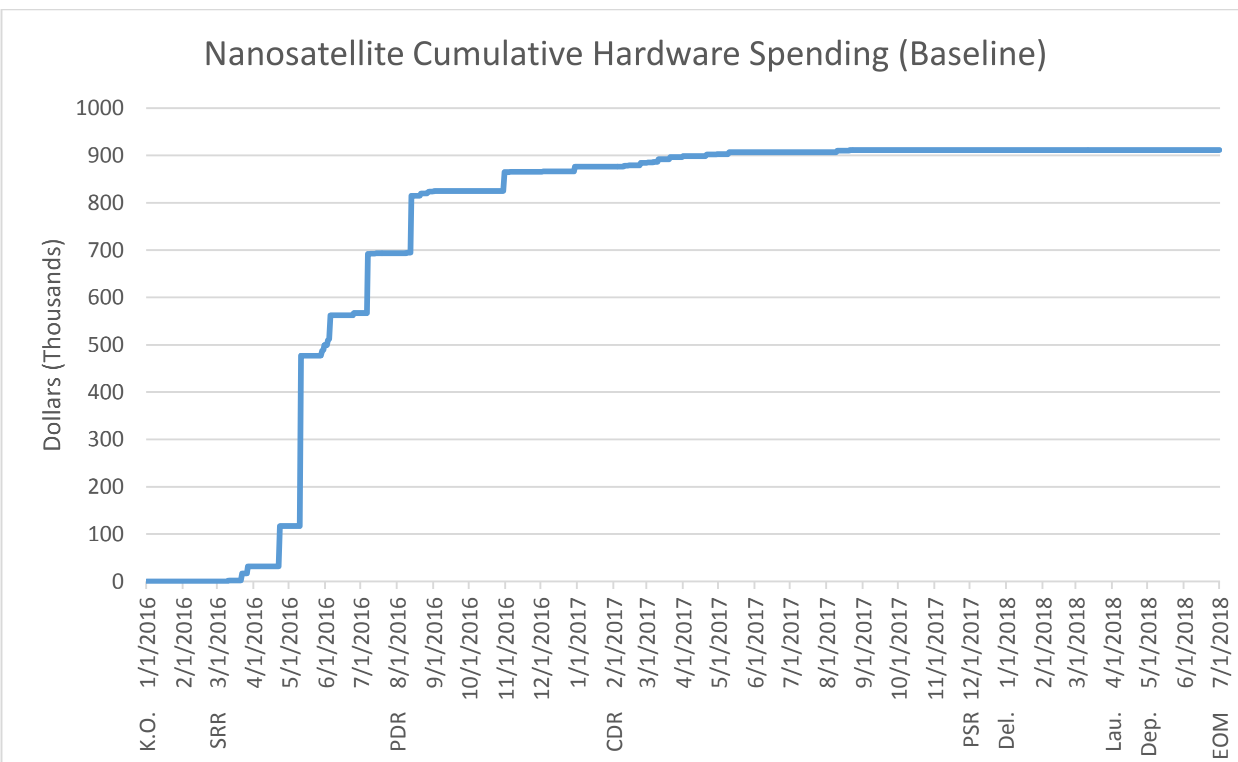


Figure 4-5: Nanosatellite Cumulative Hardware Spending (Baseline)

The total spending on hardware for this program reaches roughly $900,000. This program is unique in that a majority of the big purchases were made earlier in the program than what might be seen in other similar programs. The reason for this is that MiRaTA is the second in a series of 3U weather-sensing CubeSats, which utilizes many of the same components as its predecessor. It is interesting to point out that of this spending, over 70% is allocated to the payload, with less than 30% going to the bus. For a 3U CubeSat mission such as MiRaTA, the more complex (and significantly more expensive) payload is integrated with a relatively inexpensive bus, comprised mostly of commercial off-the-shelf (COTS) components.

4.2.3 Testing Cost

For MiRaTA, the majority of environmental tests are conducted after CDR, at which point final models of custom engineering components were finished and available for testing. The same tests are then conducted again on the flight components towards the end of the program in order to fulfill the launch provider’s secondary payload integration requirements.

For this analysis, the cost of testing is defined as the daily man-hour cost of preparing for, running, and analyzing the results of tests. For simplicity’s sake, an average two week time period is used as the amount of time it takes to go through this entire process for a particular test. The cost of testing equipment (testing rigs, development boards, and ground control stations) has already been included within the Ground Support Equipment (GSE) division of hardware cost.

The testing profiles follow those outlined in Section 2.4.2. Thermal testing is conducted five times over the course of the program. After components are tested in a room temperature environment, it is necessary to ensure that they continue to perform as expected during a sweep through their operational temperature limits, and after being stressed to their survival limits. This testing is conducted on each of the two payloads, on the integrated bus, on the radio alone as a part of a thermal vacuum test, and on the integrated space vehicle as a part of another thermal vacuum test. For the radio and during the space vehicle portion of the thermal testing, vacuum testing occurs concurrently as the tests were conducted in a thermal vacuum chamber. Each payload thermal test requires three senior staff members, due to the complexity of the payloads on this project. The integrated bus testing requires one senior staff member for supervision and seven upper-level students (one each from each student division of the project: systems engineering, communications, power, avionics hardware, avionics software, ADCS, and thermal/structures). The radio thermal vacuum testing, due to its increased complexity, requires the same personnel numbers as the bus testing, with the addition of two more senior staff members to aid in supervision and test operation. The integrated space vehicle thermal vacuum testing requires this same amount of personnel support.

Vibration testing is conducted once on the engineering component-equivalent for the space vehicle’s structure, the mass mockup, and then again for the integrated space vehicle prior to delivery. Each vibration test requires two senior staff members (one for supervision and one structures expert familiar with test operation) and all upper-level graduate students, to be responsible for each of their respective subsystems. Shock testing is also conducted on the integrated space vehicle in order to verify survivability within the unique shock envelope of the mission’s launch vehicle. This testing requires three senior staff members and two upper-level graduate students (systems engineer and structures/thermal engineer).

Radiation testing is conducted on the avionics hardware components to ensure total dosage survivability over the course of the satellite’s mission, which requires one senior staff member and two upper-level graduate students (avionics hardware and software). The personnel costs required for each of these tests are summed over the course of the program and are depicted in Figure 4-6. The costs of the individual tests are then summed together, yielding the cumulative spending chart for testing (Figure 4-7).

The total cost of environmental testing for this project is around $200,000. Of this, thermal testing makes up the greatest percentage at 44%, followed by vacuum testing at 24%, vibration testing at 21%, shock testing at 7% and radiation testing at 4%. These values directly correspond to the amount of times these tests are conducted and how much personnel support is required for each.

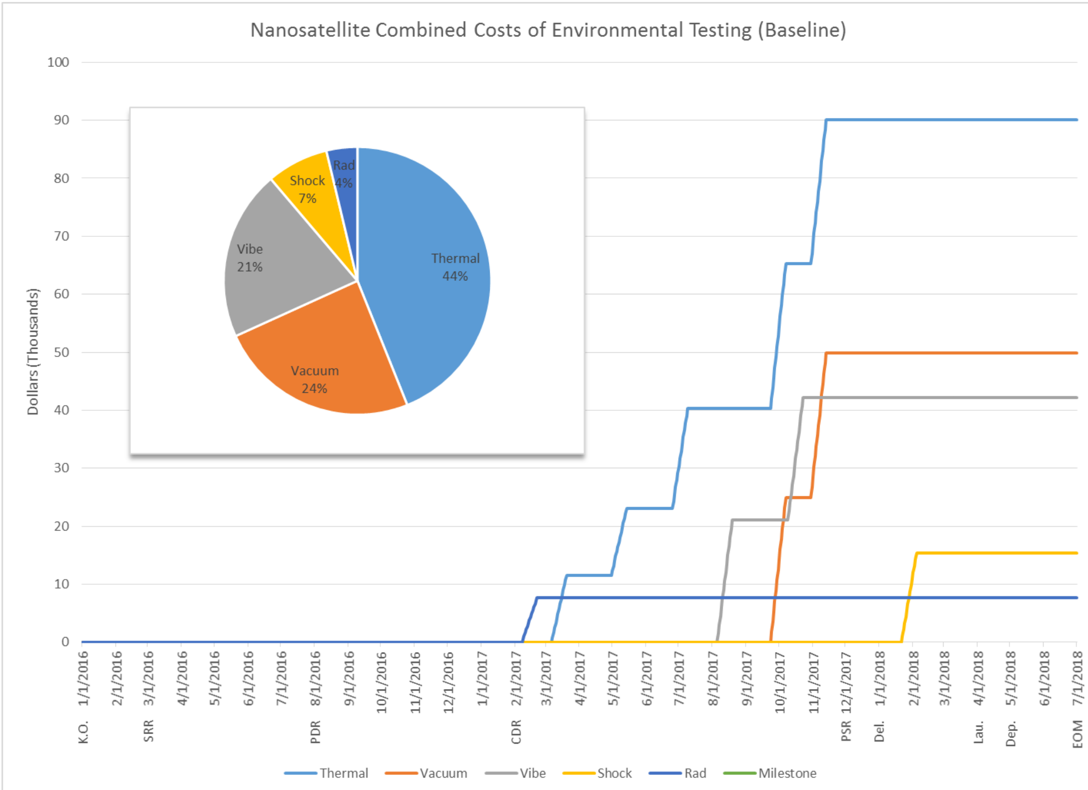


Figure 4-6: Nanosatellite Combined Costs of Environmental Testing (Baseline)

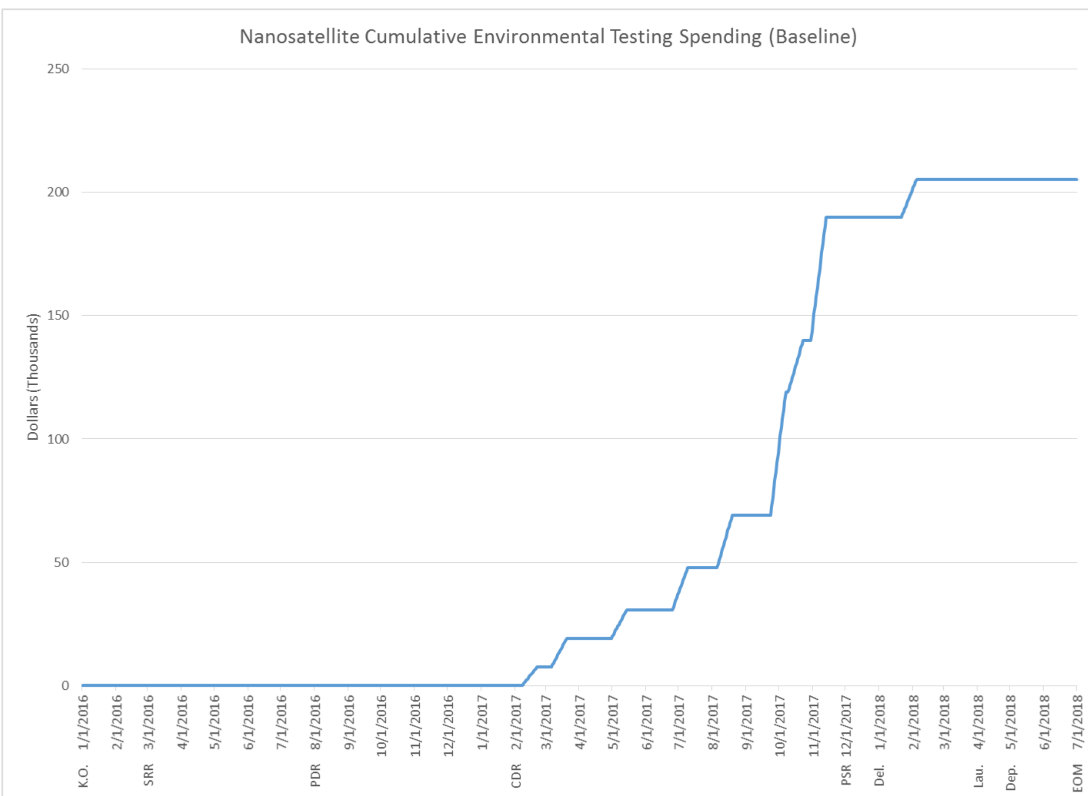


Figure 4-7: Nanosatellite Cumulative Environmental Testing Spending (Baseline)

4.2.4 Launch Cost

This portion of the total program cost covers the cost of the launch and accompanying launch integration services. Nanosatellites can compete for launch opportunities as secondary payloads onboard launch vehicles for previously-planned missions which can accommodate the additional small payloads. Depending on the nature of the satellite project, the actual cost of the launch may or may not have to be paid by the team; in the case of CubeSats participating in a space-agency funded program, they can propose to have the cost covered by the sponsor of the program. This is sometimes the case with university satellite groups, such as MiRaTA, which is getting a sponsored ride through the NASA Educational Launch of Nanosatellites (ELaNa) 14 mission. For the purpose of this analysis, which covers the total cost of the satellite program regardless of who pays for it, the cost of the launch will be included. The cost value of launching a CubeSat varies based on the launch provider. Information about the relative cost of adding a CubeSat as a secondary payload through sponsored educational rideshare opportunities is difficult to determine, and is not published publicly. Paid launch services for CubeSats offer insight into this cost, however. Spaceflight Incorporated offers rideshare opportunities at $295K per 3U CubeSat [43]. As CubeSats continue to grow in popularity within the space industry, organizations are now beginning to offer dedicated launch services for CubeSats. The NASA Launch Services Enabling eXploration and Technology (NEXT) program aims to offer a dedicated launch service for CubeSats at a cost of $300K each [44] [45]. Commercial organizations Firefly and Virgin Galactic are currently developing launch vehicles for dedicated CubeSat launches, though their prices have not yet been publicly disclosed [12] [16]. An additional organization, Rocket Lab, is currently taking reservations for its dedicated CubeSat launch vehicle, Electron, with prices starting at $200K per 3U CubeSat [14]. For the purposes of this analysis, the most conservative value for a launch currently available for 3U CubeSats, $300K, will be used.

4.2.5 External Operator Cost

The cost of external operators is the cost to pay those individuals in separate ground stations hired to aid with operational tasks such as tracking, commanding, site maintenance, fault debugging, and data analysis. This cost begins upon deployment and ends at the End of Mission (EOM). For this analysis, a daily cost of $200 is used for each external operator, and two external operators are said to be in use.

4.2.6 Summary

All of the cost categories for the program are then combined in order to gain a visual comparison of their values (Figure 4-8). Note that the personnel cost attributed to environmental testing is subtracted from the staffing cost in this depiction, to avoid double-counting. From this chart, it becomes clear that the greatest expenditure on this program is not the cost of the components that go into building the satellite; rather, it is the cost to pay the team members to build it.

The various cost categories are then summed together to yield the cumulative program spending (Figure 4-9). One time period of the greatest spending takes place about halfway between SRR and PDR, when the most expensive hardware components are ordered. From PDR to PSR, the spending is fairly steady, with the main cost drivers being the cost of staffing, and additional less-expensive hardware purchases which are made at fairly regular intervals. A spending spike is again seen at the time of launch, and then the cost of the remaining team staff and external operators brings the spending to its completion through EOM. The total amount spent on this program is roughly $3.3 million. Of this, staffing cost (excluding that associated with environmental testing) makes up 56%, the personnel cost of environmental testing makes up 6%, hardware cost makes up 28%, and the combined cost of the launch and external operators makes up 10%. It is interesting to note that the cost of staffing alone makes up the majority of the cost of a program such as this.

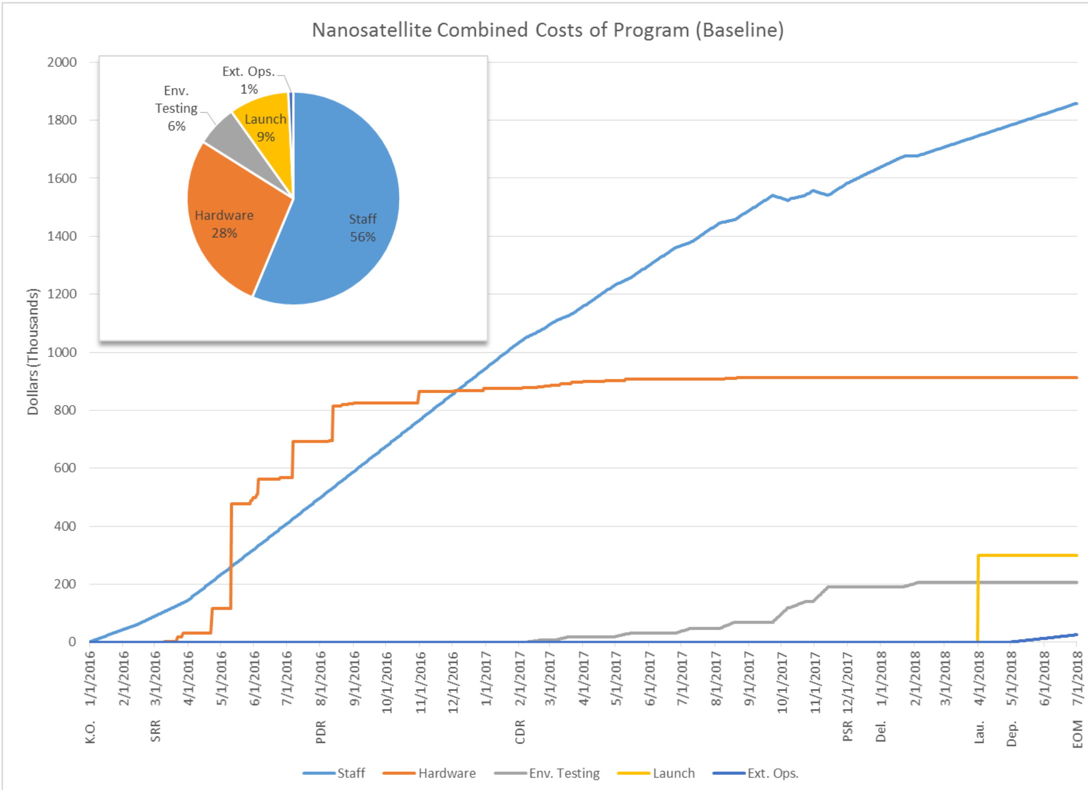


Figure 4-8: Nanosatellite Combined Costs of Program (Baseline)

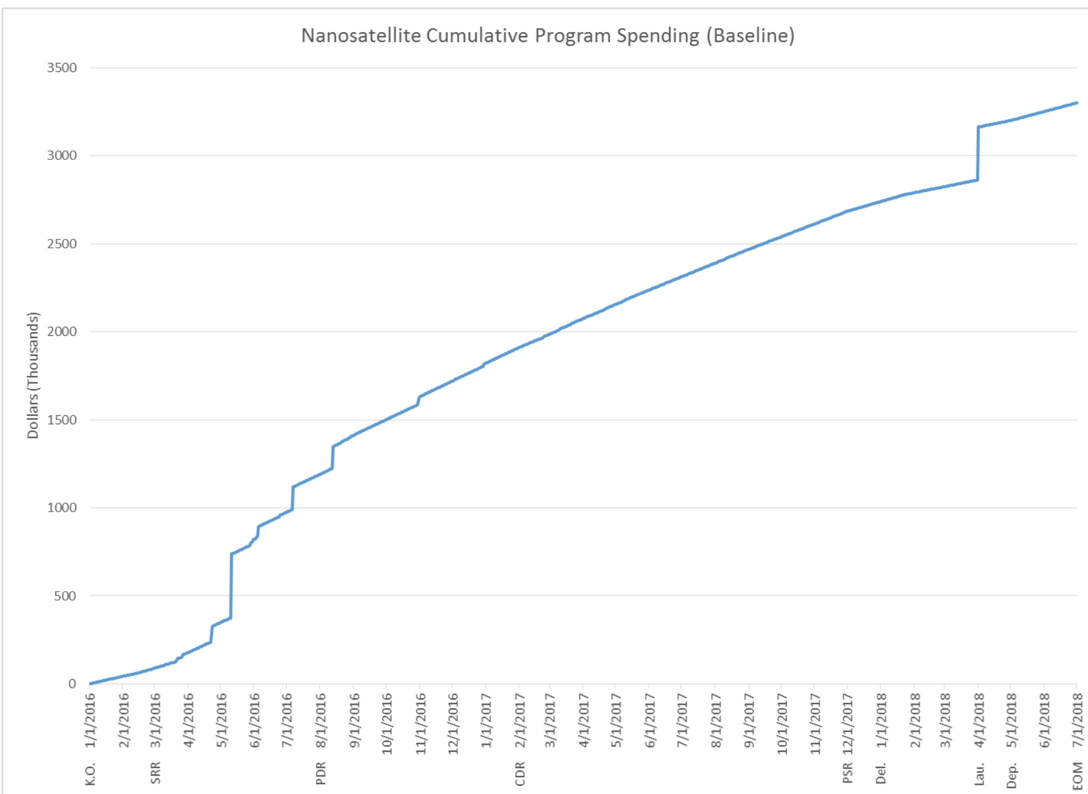


Figure 4-9: Nanosatellite Cumulative Program Spending (Baseline)

4.3 Two Satellites (Unplanned Failure First Iteration) – 2:1 λ U

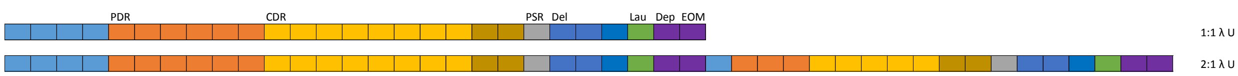
Next, we analyze the total cost of a program which fails on its first attempt and then decides to build another satellite for a second attempt. The inspiration behind this analysis is the MicroMAS to MicroMAS-2 project history, wherein MicroMAS had an unrecoverable failure on-orbit and MicroMAS-2 was built to meet unaccomplished mission objectives from its predecessor. With the thought that after having already built MicroMAS the first time, building a second should be easier and faster, MicroMAS-2 is intended to be an ambitious program, with a kick-off to delivery time of only one year. The same spacecraft design is used, with slight modifications to component selection and software to prevent a repeat failure.

The total program cost for this second attempt is modeled in the same manner as before. This mission architecture, which takes two satellite iterations to achieve mission success with an unplanned failure on its first iteration, is designated "2:1 λ U," where "λ" corresponds to the attempt:success ratio and "U" corresponds to the unplanned failure. In order to be able to accurately compare to the baseline, this analysis looks at what the total cost for the MiRaTA program would be if it went through the same two-satellite mission profile as MicroMAS, in which the first is an unplanned failure. The data for this analysis is again taken from the MiRaTA project, with representative values for staffing, hardware cost, testing regimes, external operators and launch cost from the program being used.

For MicroMAS-2, the time from kickoff to delivery is half that of MicroMAS-1 and MiRaTA, at 12 months. The thought behind this short duration is that, since it has been done before, it should take about half the time to do it again. Certain steps are skipped the second time around, such as the SRR, because the design and requirements of the second iteration spacecraft are the same as the first. The post-delivery time, driven by the launch provider integration time and the mission duration is the same, at 6 months. This gives the program duration of the second-iteration satellite (henceforth to be called "Satellite 2") of 18 months, with a total program duration of 48 months. Table 4-3 provides a summary and graphical depiction of the program duration breakdown by milestone for this two-satellite iterative mission.

Table 4-3: Nanosatellite Program Duration by Milestone (2:1 λ U)

Milestone	Kickoff	SRR	PDR	CDR	PSR	Delivery	Launch	Deploy	EOM
	Satellite 1								
Time until next (months)	2 mo.	5 mo.	6 mo.	10 mo.	1 mo.	3 mo.	1 mo.	2 mo.	End
	Time to delivery: 24 mo.					Post-delivery time: 6 mo.			
	Satellite 1 duration: 30 mo.								
	Satellite 2								
Time until next (months)	-	1 mo.	3 mo.	7 mo.	1 mo.	3 mo.	1 mo.	2 mo.	End
	Time to delivery: 12 mo.					Post-delivery time: 6 mo.			
	Satellite 2 duration: 18 mo.								
	Total program duration: 48 mo.								



4.3.1 Staff Cost

The staff cost is modeled the same as before. The project members are divided into three categories based on their cost: senior staff members, engineering staff, and additional staff (faculty and engineer support, upper-level students and lower-level students for a university-based program, respectively). The same staffing numbers are used between each milestone as the single-satellite mission profile during the second iteration. Note that on the second build, there is no SRR (as the requirements have not changed from the first mission); for this reason, the staffing numbers (and the corresponding cost) during the time period between kickoff and PDR for the second build will be set equal to the time period between SRR and PDR for the first build. First, the staffing numbers are tracked and plotted over the course of the program (Figure 4-10). Note that Kickoff 2 (K.O. 2) corresponds with, and occurs at the same time as, the End of Mission for the first satellite (EOM 1).

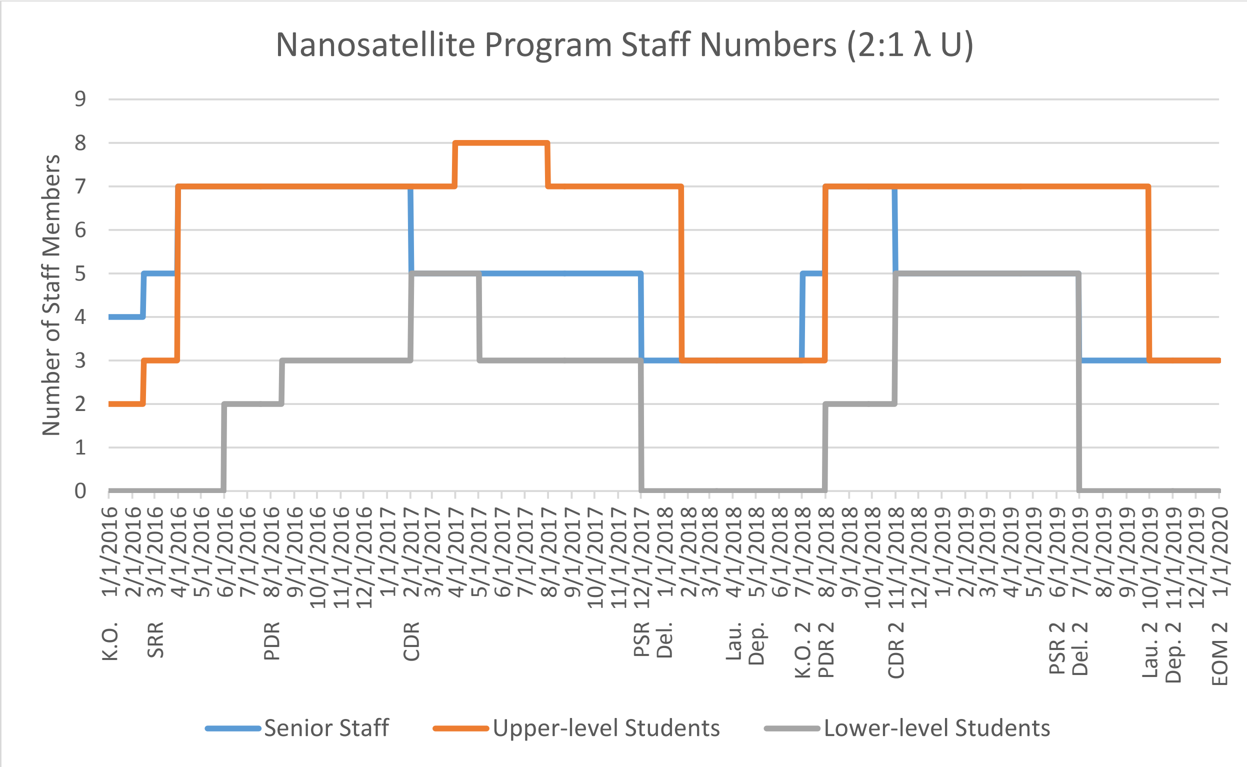


Figure 4-10: Nanosatellite Program Staff Numbers (2:1 λ U)

Next, the staffing numbers are multiplied by their daily cost values (same as before, see Table 4-2) to yield the chart for daily staffing spending (Figure 4-11). The daily staffing spending is then integrated to determine the cumulative amount spent over the course of the program (Figure 4-12). The total spending on staffing comes out to be roughly $3,244,000 for this program, an increase of $1,180,000 from the $2,064,000 total staffing cost for the single-iteration satellite program discussed in Section 4.2. This means that the staffing cost to build the second satellite is 57% of that for the first satellite.

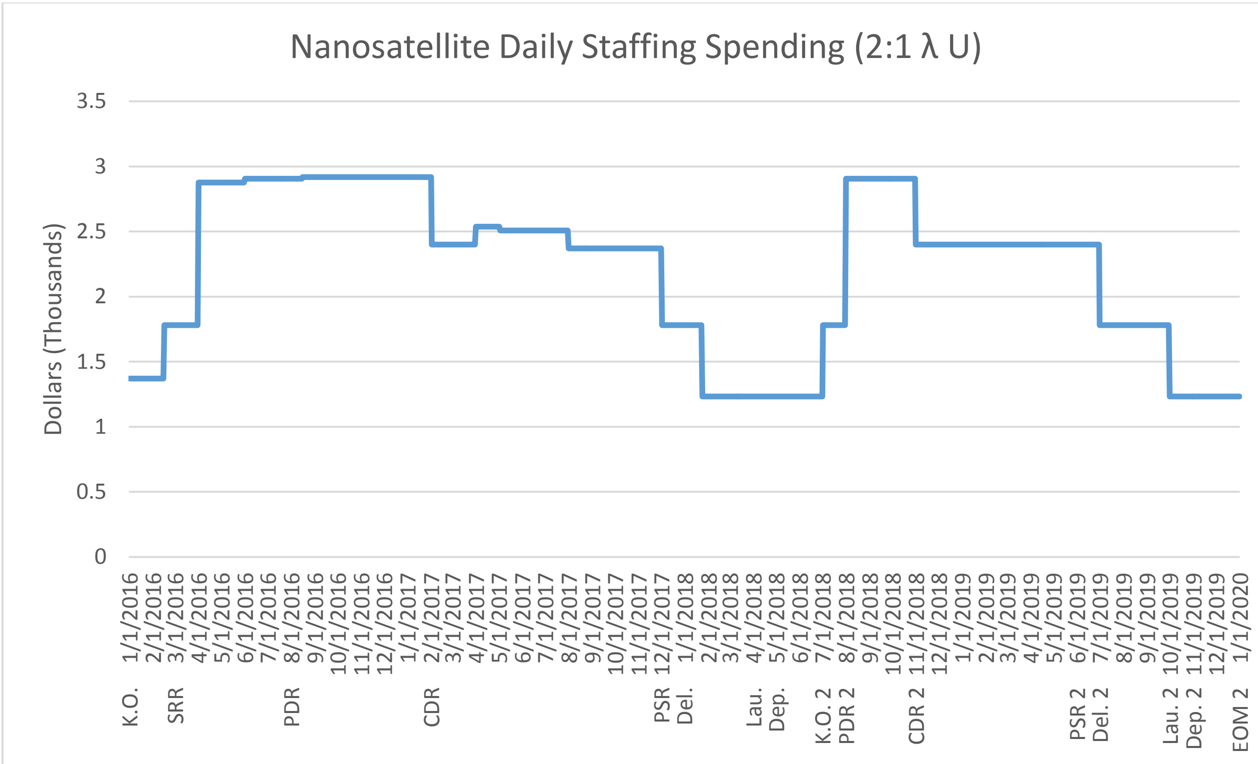


Figure 4-11: Nanosatellite Daily Staffing Spending (2:1 λ U)

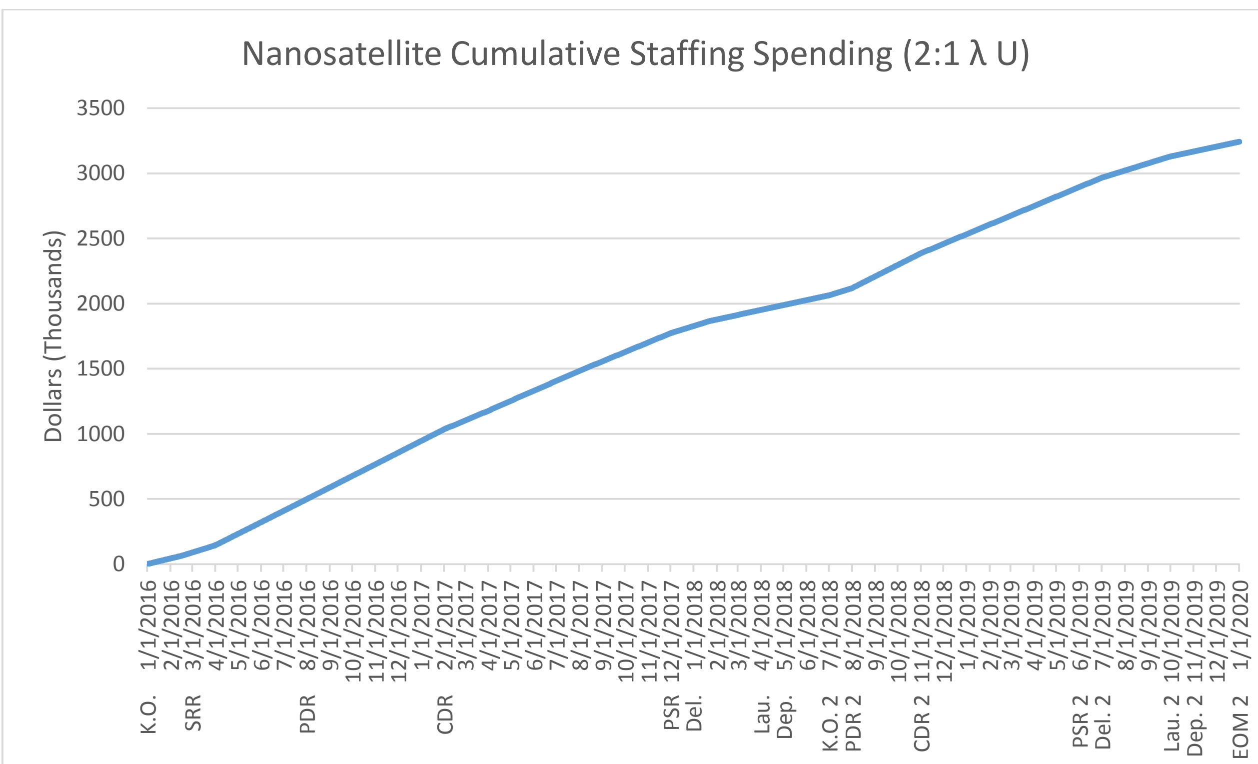


Figure 4-12: Nanosatellite Cumulative Staffing Spending (2:1 λ U)

4.3.2 Hardware Cost

The total cost of hardware for this two-iteration satellite program is then analyzed. The following assumption is made for this analysis: all flight and engineering model hardware need to be purchased again the second iteration, but there is no need to repurchase any ground support equipment. Figure 4-13 shows the daily spending on bus flight and engineering hardware, as well as ground support equipment over the course of the program.

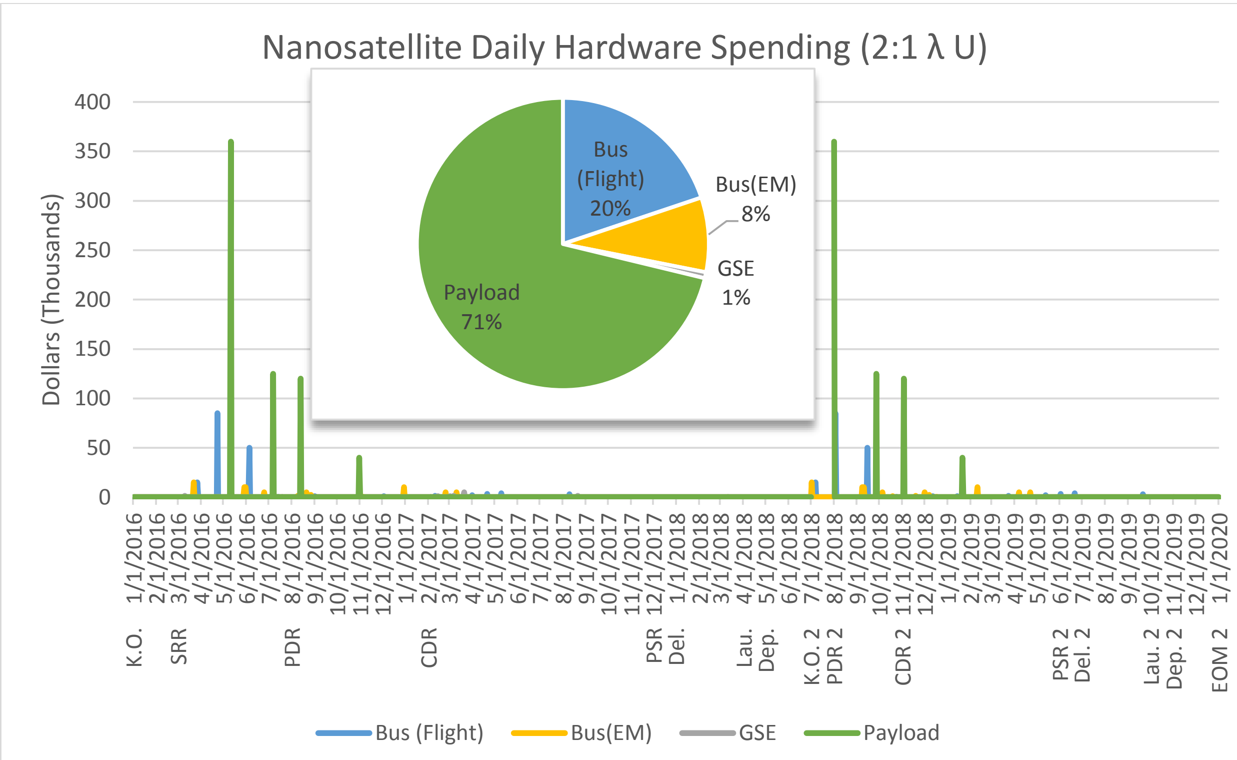


Figure 4-13: Nanosatellite Daily Hardware Spending (2:1 λ U)

Like the hardware spending for the single-satellite program, over 70% is allocated to the payload, with less than 30% to the bus. While the bus costs significantly less than the payload, it is by no means less important; it is important to remember that the success of payload operations is contingent upon the proper functionality of the bus. Next, the daily spending on hardware over the course of the program is summed, yielding the cumulative spending chart shown in Figure 4-14.

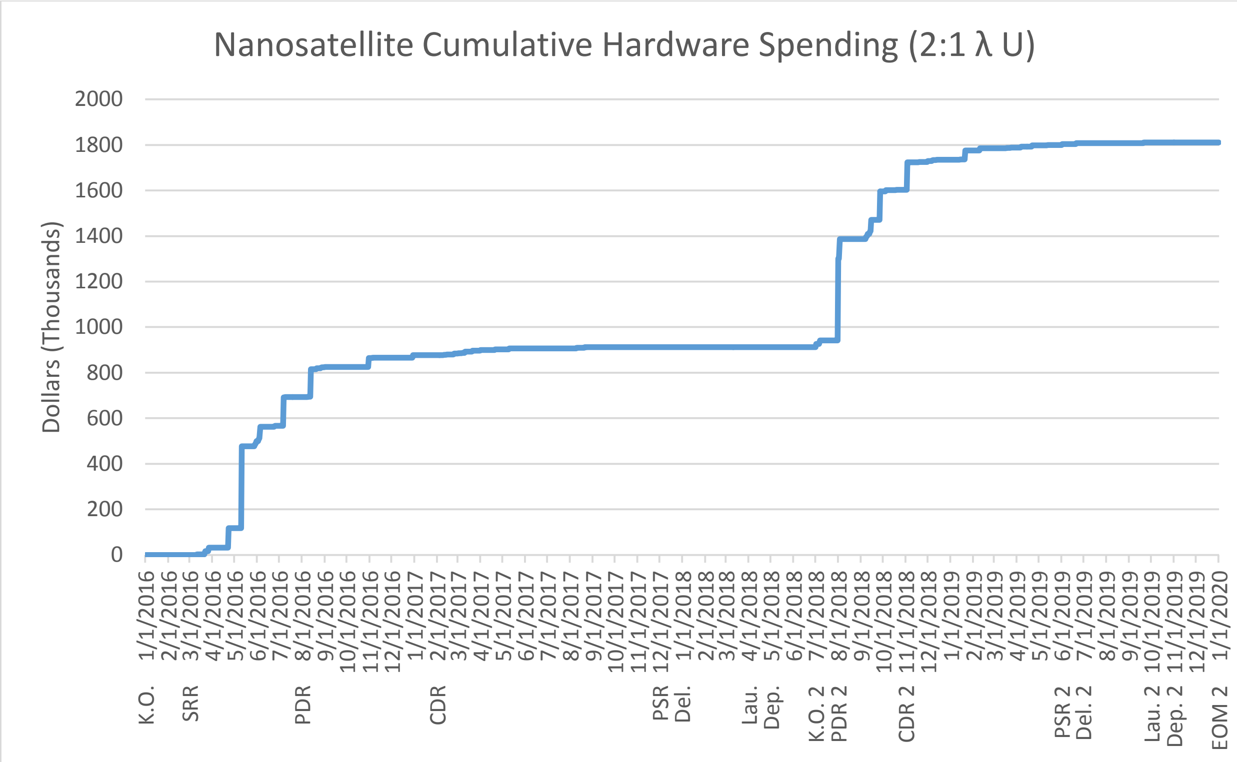


Figure 4-14: Nanosatellite Cumulative Hardware Spending (2:1 λ U)

For each iteration, the majority of hardware spending occurs between SRR and PDR; this is due to the fact that for both, the components that would be required were defined early in the program, and ordering early is a risk-mitigation strategy to combat long lead times and late product deliveries from vendors. As would be anticipated, the cost of hardware for the second iteration is nearly equal to that of the first iteration; the second-iteration satellite's hardware costs 98% of the first-iteration, as the only savings gained between each iteration is on ground support equipment, which makes up a meager 1% of the total hardware cost.

4.3.3 Testing Cost

The cost of testing is tracked as before, by attributing the cost of total man-hours spent planning, running, and analyzing environmental tests. An assumption is made that the same amount of testing is conducted for the second-iteration satellite as the first, simply in a more compressed time period. Figure 4-15 shows the combined costs of environmental spending over the course of the program and Figure 4-16 shows the cumulative environmental testing spending.

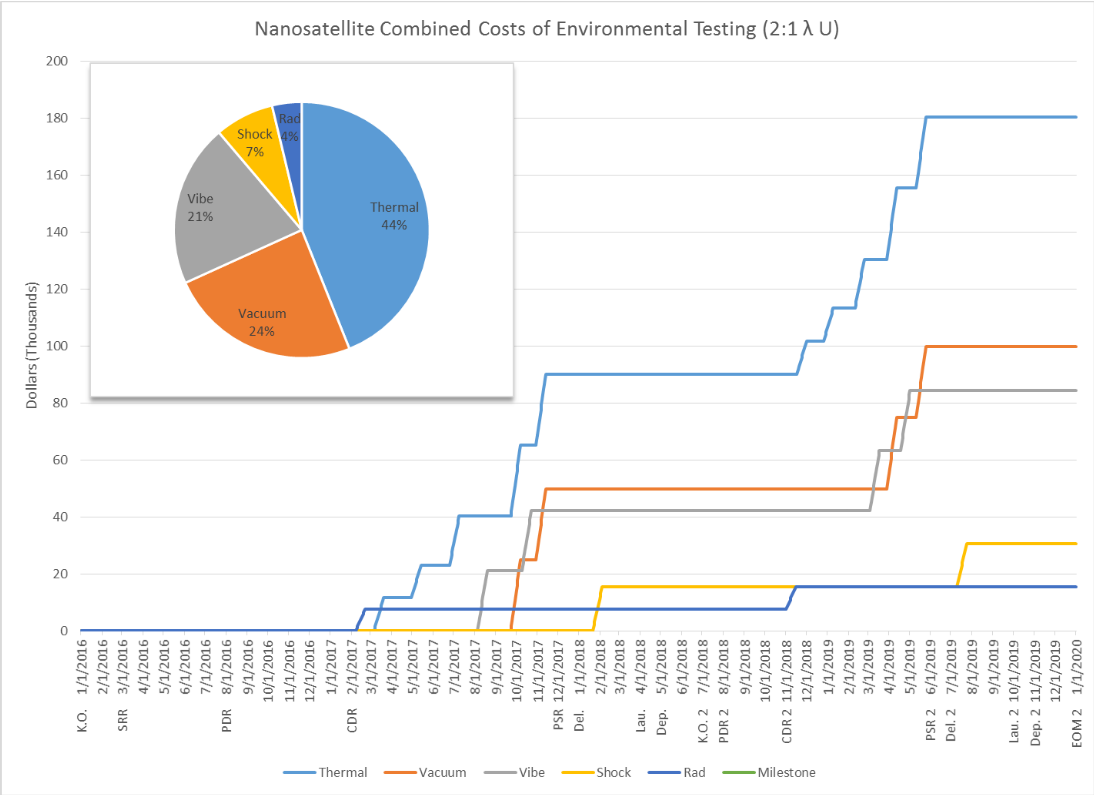


Figure 4-15: Nanosatellite Combined Costs of Environmental Testing (2:1 λ U)

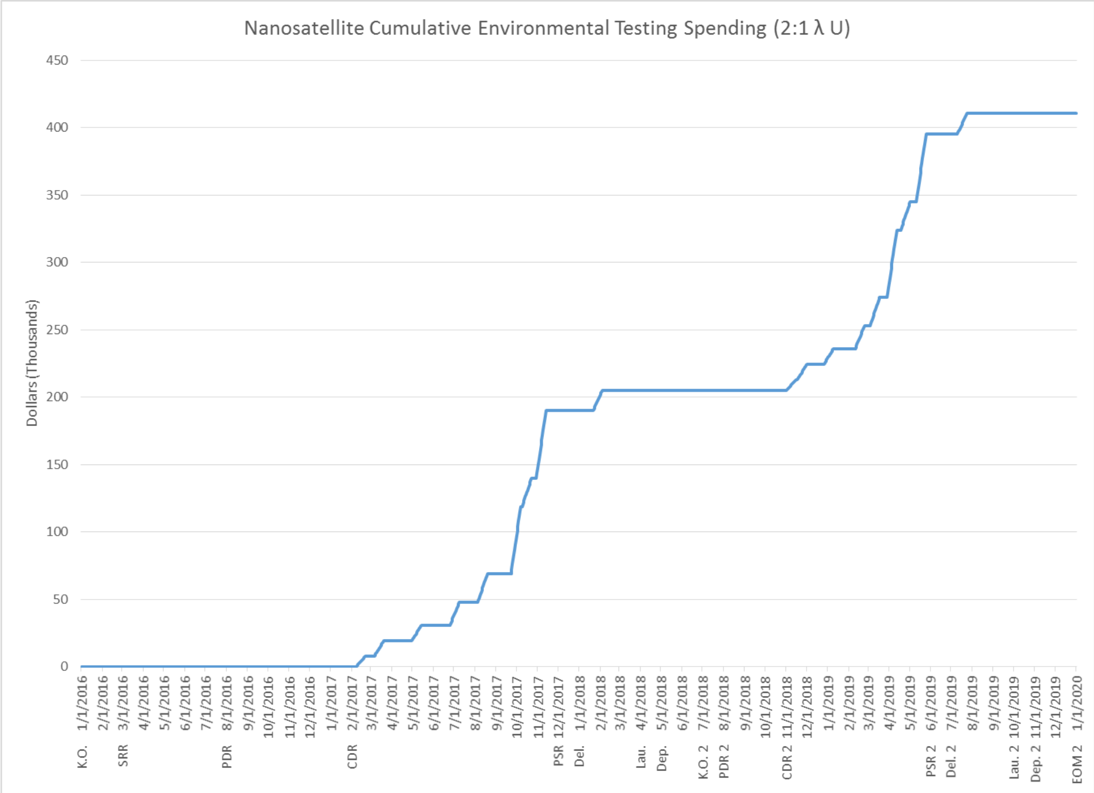


Figure 4-16: Nanosatellite Cumulative Environmental Testing Spending (2:1 λ U)

The total cost of environmental testing for this project is around $400,000. Since the same tests are conducted on the second iteration as on the first, the percentage distribution of the tests is the same: thermal testing makes up the greatest percentage at 44%, followed by vacuum testing at 24%, vibration testing at 21%, shock testing at 7% and radiation testing at 4%. These values directly correspond to the amount of times these tests are conducted and how much personnel support is required for each.

4.3.4 Launch and External Operator Cost

The cost of launch and external operators are then added. There are now two launches that add to the total program cost, as well as two separate operational regimes which require the assistance of external personnel. A $200 per day per operator cost with two operators in use during the period from deployment to EOM is again assumed. The launch cost (plus launch integration services) is kept at $300,000 for each launch.

4.3.5 Summary

All of the cost categories for the program are then combined in order to gain a visual comparison of their values (Figure 4-17). As before, the personnel cost attributed to environmental testing is subtracted from the staffing cost in this depiction, to avoid double-counting. This chart again emphasizes the fact that staffing is the largest expenditure in a program.

The various cost categories are then summed together to yield the cumulative program spending (Figure 4-18). The total amount spent on this two-iteration program is roughly $5.7 million. Of this, staffing cost (excluding that associated with environmental testing) makes up 50%, the personnel cost of environmental testing makes up 7%, hardware cost makes up 32%, and the combined cost of the launch and external operators makes up 11%.

Note that after factoring in the second attempt, the cost of hardware now makes up a greater percentage of the total cost than before, with the personnel cost subsequently decreasing slightly. This is due to the fact that the same amount is spent during the second iteration on hardware as the first, though the duration of the second iteration is shorter than the first, corresponding to a decrease in the amount of time for which the personnel must be paid.

This CubeSat program, with a complex payload and a 24-month program duration from kickoff to delivery, followed by a 6-month post delivery period until the end of the mission, costs $3.3 million total. In the case where the satellite fails to meet its mission objectives on the first attempt and then the project team decides to rebuild the satellite for a second attempt, the cost of the second iteration is approximately $2.4 million, when this second build has a kickoff to delivery time of 12 months. The cost of the second satellite is approximately 73% of the cost of the first satellite; what this says is that a program whose CubeSat fails on the first attempt spends 27% less building it a second time, assuming that a second build only takes half the time as the first.

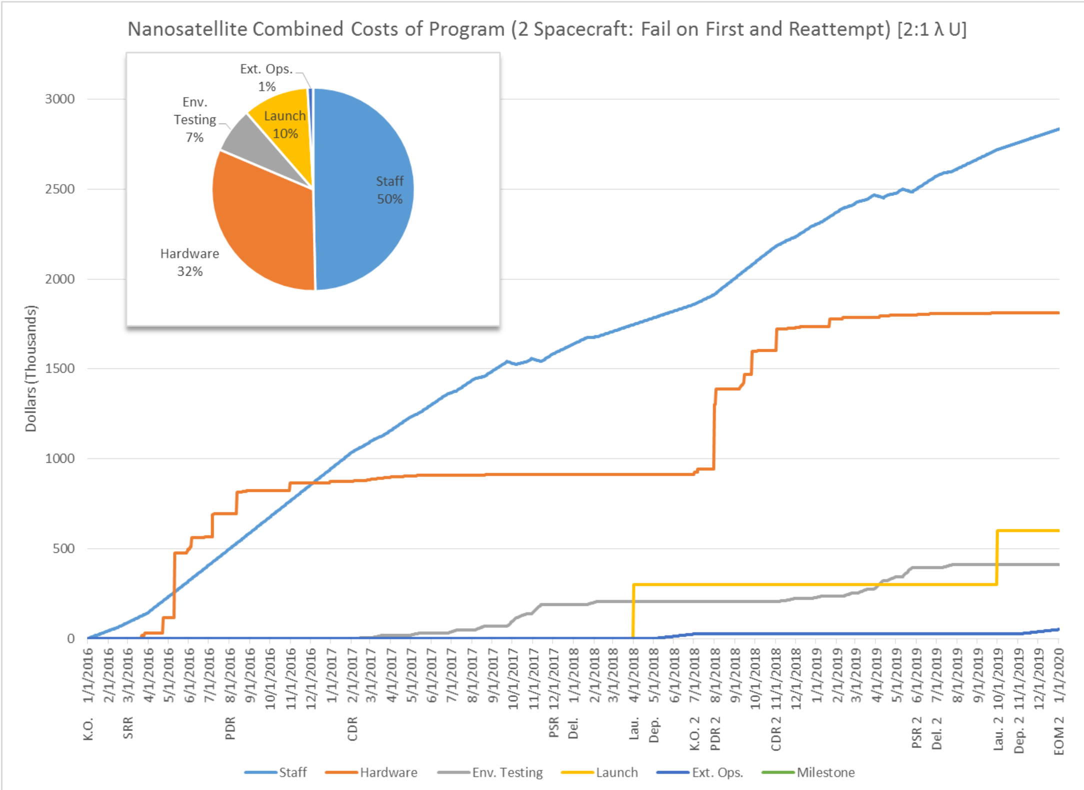


Figure 4-17: Nanosatellite Combined Costs of Program (2:1 λ U)

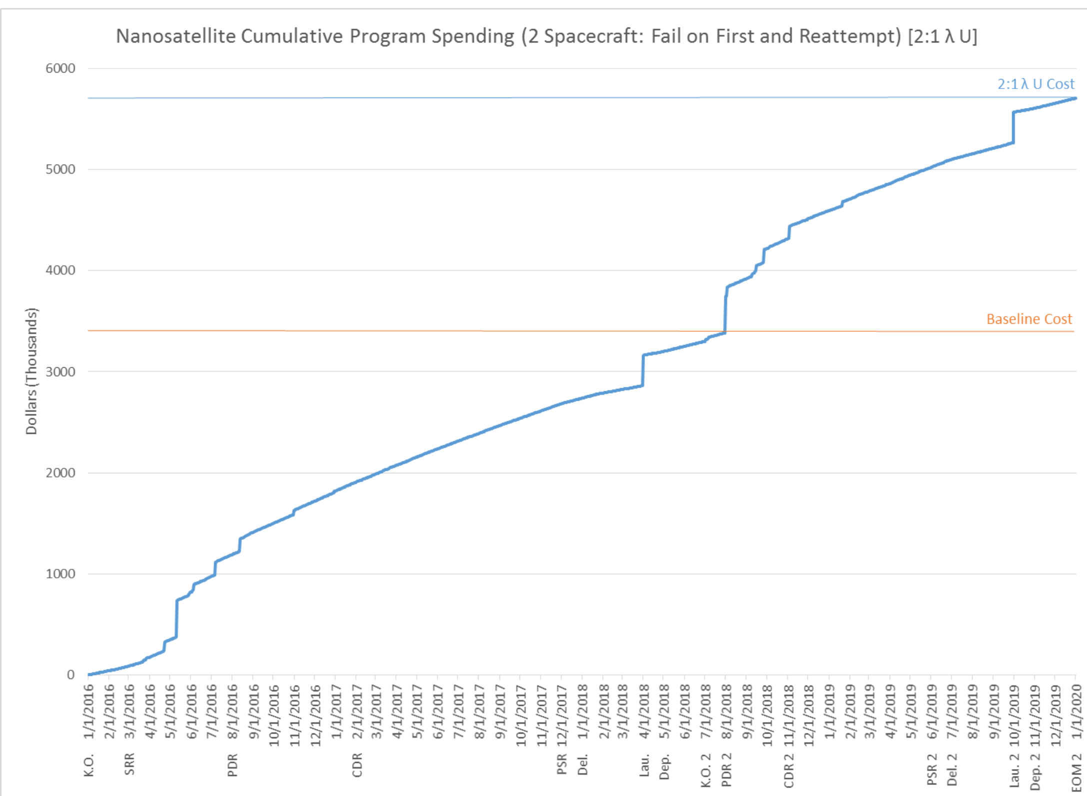


Figure 4-18: Nanosatellite Cumulative Program Spending (2:1 λ U)

Chapter 5: CubeSat Beta Build Strategies

5.1 Overview

We make the case, using the CubeSat beta build strategies, that planning to build two satellites from the beginning increases a CubeSat team's probability of success, and does so at a lower cost than if the CubeSat team decided to build a second satellite only after having their first fail. We update testing and build parameters in our models to reflect the cost of programs using the beta build strategies. For this analysis, three beta build strategies, which cover some of the largest parameter variations across strategies, are discussed:

- Beta Build Strategy 1: The "baseline" beta build strategy, in which the program decides to build two flight satellites from the beginning.
- Beta Build Strategy 2: The program still decides to build two satellites from the beginning, though does not integrate the expensive payload on the first spacecraft.
- Beta Build Strategy 3: This strategy maintains the assumptions of the first two strategies, but in this strategy, the first spacecraft is not launched at all. Instead, testing is lengthened for both on the ground. It is still assumed that the first spacecraft is completely expendable.

The philosophy behind each of these strategies are discussed, and the total program cost of each are analyzed as before. It should be noted that the purpose of this analysis is to compare the relative differences between a baseline 3U CubeSat program, one which fails a first time and succeeds after a second, and programs which utilize these alternate build strategies; specific cost values are not intended to serve as quotes for prospective CubeSat engineering teams, as these values will naturally differ across programs. Since all programs vary, teams should first consider the unique aspects of their program prior to the decision to alter their program strategies.

5.2 Beta Build Strategy 1 – 2:1 λ P 1

In this alternate build strategy, the CubeSat engineering team makes the decision to build two spacecraft from the beginning. For this analysis, it is assumed that the team purchases double of all hardware components, both engineering and flight model, with the exception of ground support equipment. The primary cost-saving driver for this strategy versus deciding to build a second spacecraft only after an initial failure is the fact that, in this scenario, the team builds and tests both spacecraft simultaneously. This subsequently decreases the amount of time between the first spacecraft's launch and the second's delivery, as the spacecraft has already been built (Figure 5-1).

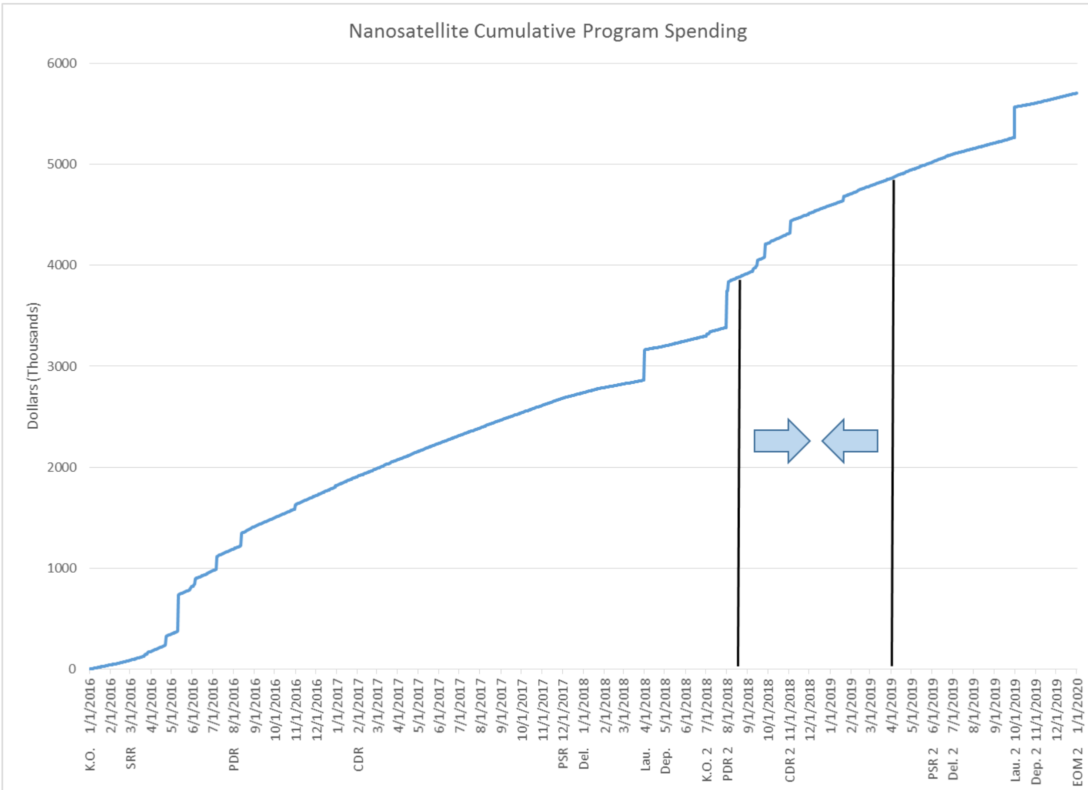
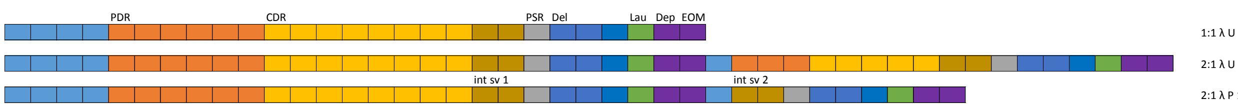


Figure 5-1: Beta Build Strategy: Effect on Two-Spacecraft Schedule

This strategy benefits from the first spacecraft mission acting as a "test on orbit," where during this time the team can analyze and diagnose any problems which may arise and fix them on the second spacecraft. Then, after EOM 1, all the team's efforts again turn to the second spacecraft to repeat final integrated testing after the modifications from the first, and deliver. Table 5-1 provides a summary of the program duration by milestone for this strategy, as well as a graphical comparison to the baseline and 2:1 λ U scenarios.

Table 5-1: CubeSat Program Duration by Milestone (2:1 λ P 1)

<table>
<tr><td>Milestone</td><td>Kickoff</td><td>SRR</td><td>PDR</td><td>CDR</td><td>PSR</td><td>Delivery</td><td>Launch</td><td>Deploy</td><td>EOM</td></tr>
<tr><td></td><td colspan="9">Satellite 1</td></tr>
<tr><td>Time until next (months)</td><td>2 mo.</td><td>5 mo.</td><td>6 mo.</td><td>10 mo.</td><td>1 mo.</td><td>3 mo.</td><td>1 mo.</td><td>2 mo.</td><td>End</td></tr>
<tr><td></td><td colspan="5">Build & test both s/c simultaneously</td><td colspan="4">Modify s/c 2 based on s/c 1 ops</td></tr>
<tr><td></td><td colspan="5">Time to delivery: 24 mo.</td><td colspan="4">Post-delivery time: 6 mo.</td></tr>
<tr><td></td><td colspan="9">Satellite 1 duration: 30 mo.</td></tr>
<tr><td></td><td colspan="9">Satellite 2</td></tr>
<tr><td>Time until next (months)</td><td>-</td><td>1 mo.</td><td colspan="2">2 mo.</td><td>1 mo.</td><td>3 mo.</td><td>1 mo.</td><td>2 mo.</td><td>End</td></tr>
<tr><td></td><td colspan="5">Time to delivery: 4 mo.</td><td colspan="4">Post-delivery time: 6 mo.</td></tr>
<tr><td></td><td colspan="9">Satellite 2 duration: 10 mo.</td></tr>
<tr><td></td><td colspan="9">Total program duration: 40 mo.</td></tr>
</table>



The total program cost is analyzed in the same manner as before. This analysis maintains the same assumptions from the 2:1 λ U analysis, with the following differences: all duplicate purchases of engineering and flight hardware for the second spacecraft now occur at the same time as the initial purchases, and all testing that takes place before the first spacecraft launches is now 1.5x longer, reflecting the additional time it will take due to both spacecraft being built and tested simultaneously. The results of the analysis are given in Figure 5-2 and Figure 5-3. Notice that, in this strategy, the relative percentage of spending on personnel decreases while that of the other categories increases. This is due to the decreased time after the first spacecraft launch which is saved by planning to build two spacecraft from the beginning. The total program cost for this scenario is roughly $5.1 million. This corresponds to 153% of the cost of the 3U baseline (1:1 λ). Recall that the scenario in which a second spacecraft is built after an initial failure (2:1 λ U) corresponded to 173% of the cost of the 3U baseline; what this means is that a program saves 20% from the 2:1 λ U scenario by planning to build two spacecraft from the beginning.

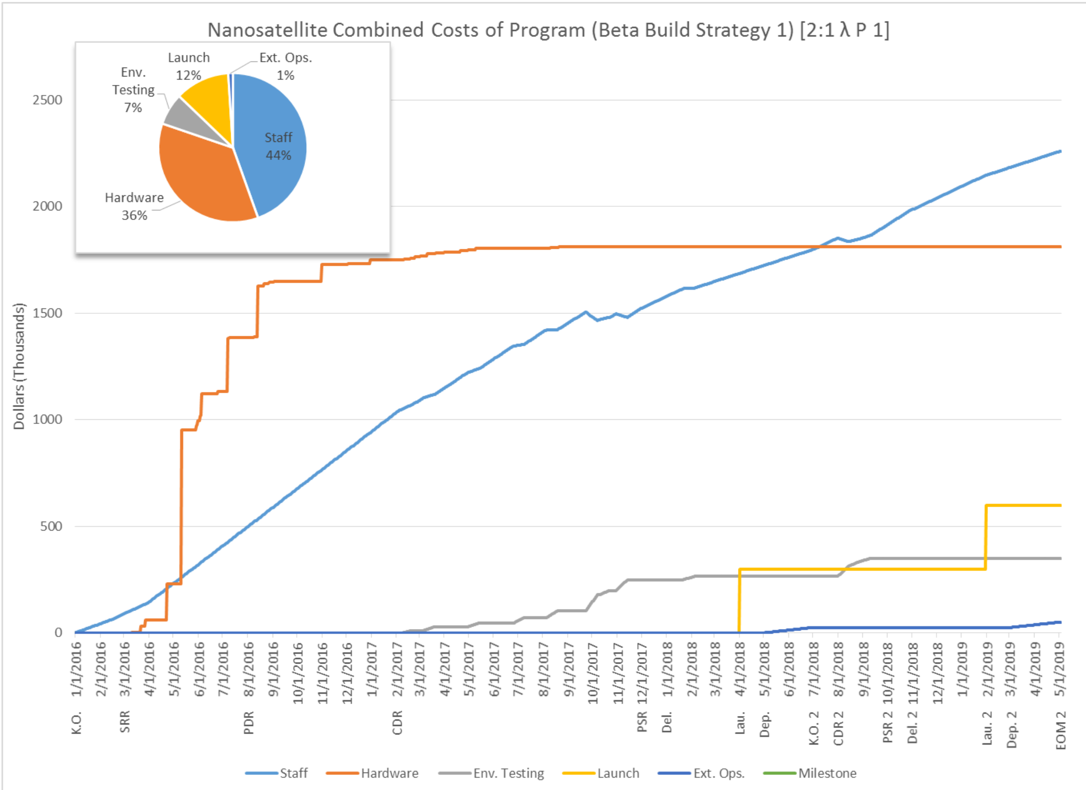


Figure 5-2: Nanosatellite Combined Costs of Program (2:1 λ P 1)

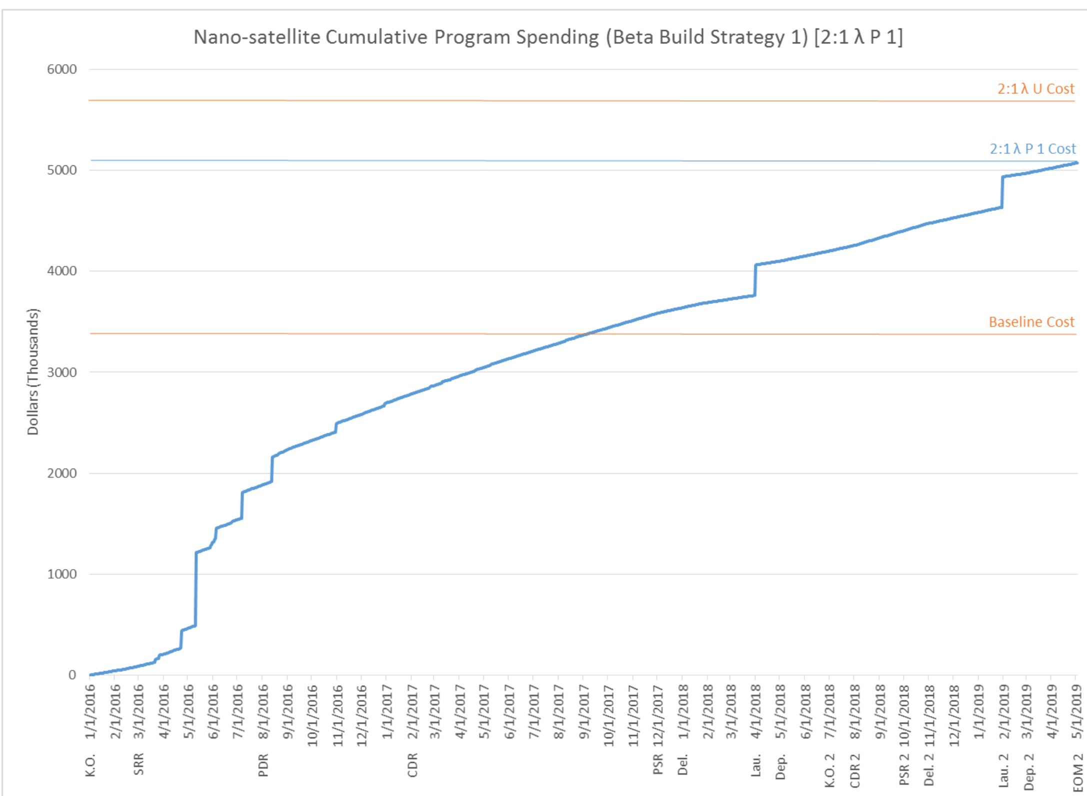


Figure 5-3: Nanosatellite Cumulative Program Spending (2:1 λ P 1)

5.3 Beta Build Strategy 2 – 2:1 λ P 2

This alternate build strategy maintains the same definitions and assumptions from the first beta build strategy, but removes the integration of the payload on the first spacecraft. Recall that only 20% of first-successfully-launched CubeSats have achieved all of their mission objectives (Figure 3-7). Also, recall the fact that the majority of CubeSat failures to date have been the result of inconsistent communication between the spacecraft and the ground station (Figure 3-2). This type of failure is completely independent of the successful operation of the CubeSat's payload. Yet with the miniaturization of technologies and the subsequent increasing complexity of what a CubeSat's payload can hold, a CubeSat's payload can now comprise the majority of the hardware cost of the satellite; this is the case with MiRaTA, for which 70% of the hardware budget is allocated to the payload (Figure 4-4). Together, these observations indicate that it may be advantageous for a satellite engineering group to not integrate a complex payload into their CubeSat if it is the first they have ever built. While they could still be successful in this situation, the odds, risks, and stakes are not in their favor. This alternate build strategy uses this line of thinking, and utilizes the first spacecraft primarily as an on-orbit verification of the integrated bus (and successful communication with the ground), while reducing cost by building only one payload which gets integrated on the second spacecraft. The analysis results are shown in the following figures (Figure 5-4 and Figure 5-5). Note that, as expected with the purchase of only one set of payload components, the overall hardware cost of the spacecraft has decreased, with the other categories now taking up a greater percentage of the total spending. The total program cost for this strategy is roughly $4.4 million. This corresponds to 133% of the cost of the 3U baseline (1:1 λ). Recall that the scenario in which a second spacecraft is built after an initial failure (2:1 λ U) corresponded to 173% of the cost of the 3U baseline, so this represents a 40% of the baseline cost saved from that scenario. There is certainly more risk associated with this strategy than 2:1 λ P 1 due to only one payload being integrated, but if the team is sufficiently confident in the proper functionality of their payload, then the amount of risk abated as compared to the baseline scenario could make the addition of one-third of the cost of the baseline program cost well worth it.

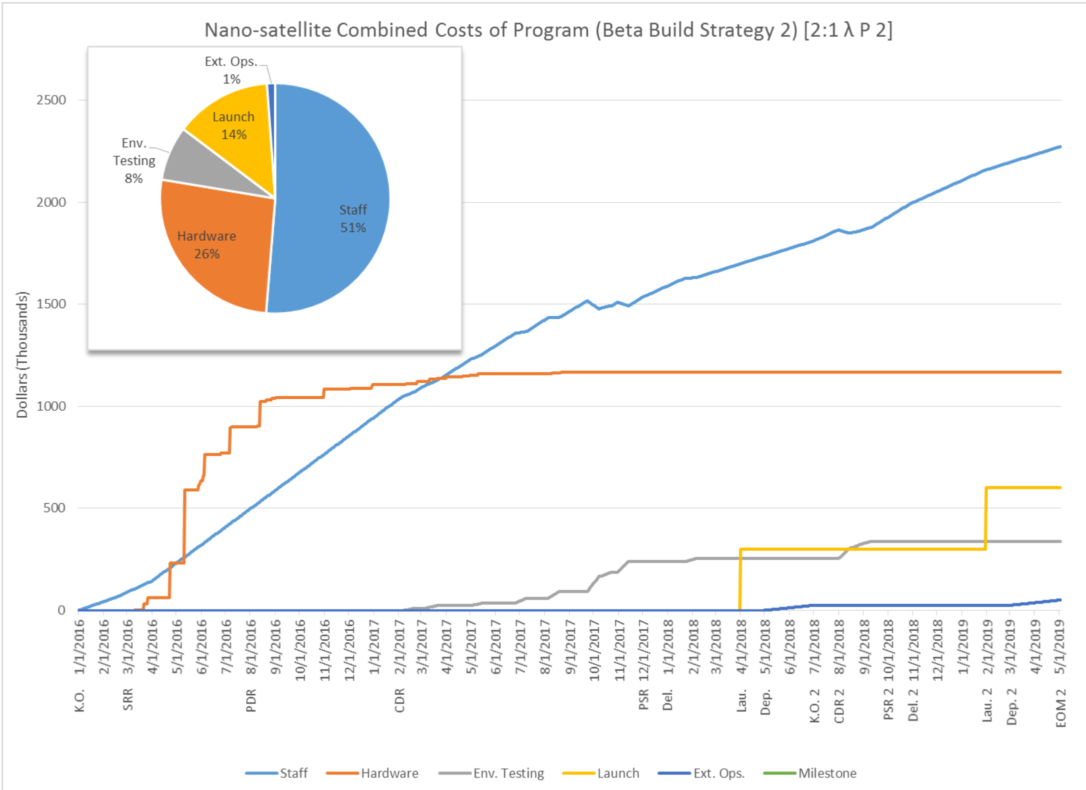


Figure 5-4: Nanosatellite Combined Costs of Program (2:1 λ P 2)

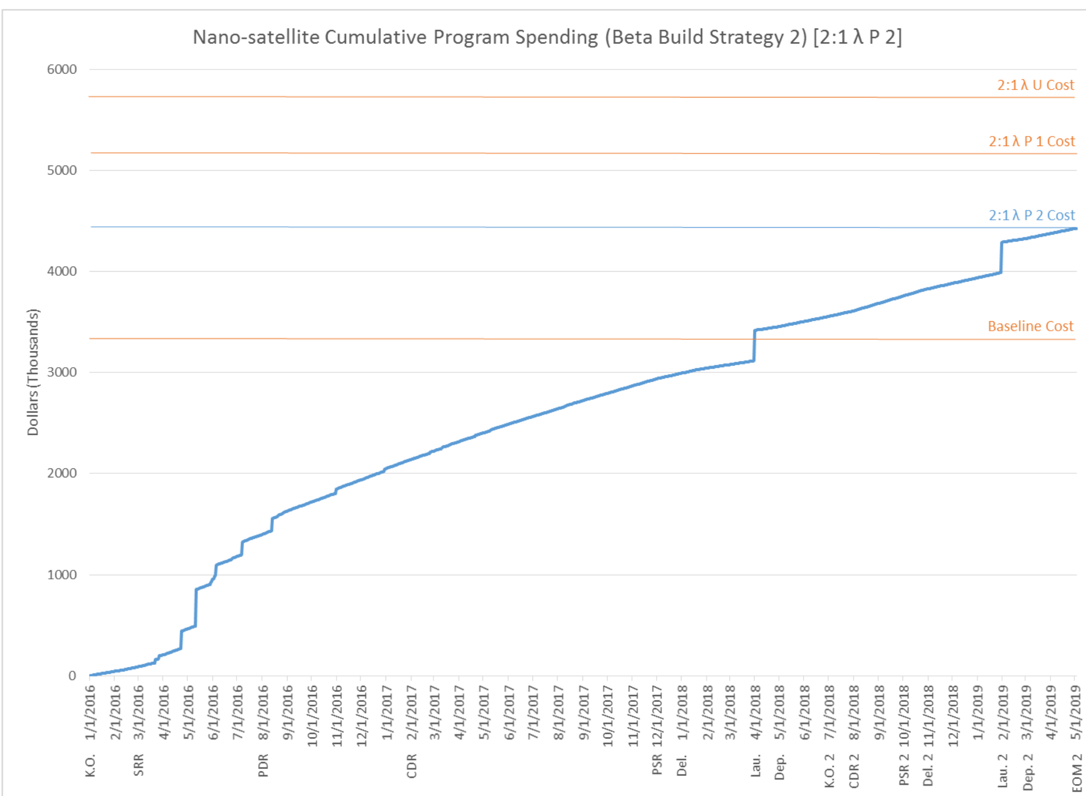


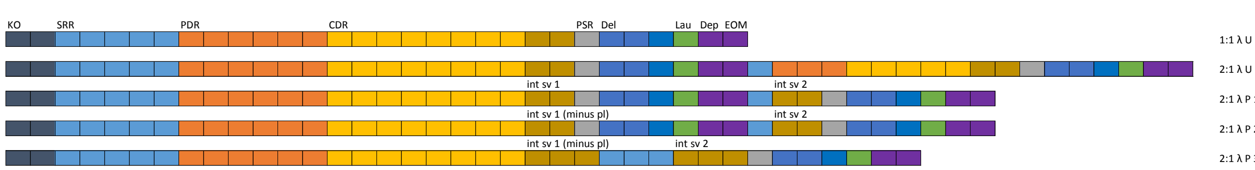
Figure 5-5: Nanosatellite Cumulative Program Spending (2:1 λ P 2)

5.4 Beta Build Strategy 3 – 2:1 λ P 3

The final beta build strategy serves as an experiment to how much one could decrease the cost of the two-satellite-build strategy. The parameters of this strategy have therefore been adjusted the most. It maintains all the assumptions of the previous strategies, but now adds the following change: the first spacecraft is not launched at all; instead, use the time that would have been previously spent on the first spacecraft's mission to lengthen the integrated testing on the ground for both spacecraft iterations. This strategy still assumes the first spacecraft is completely expendable, so two sets of all engineering and flight model components (with the exception of the payload) are still purchased. Table 5-2 provides a summary of the program duration by milestone, as well as a graphical representation of this program as compared to those previously discussed. The results of the analysis are shown in Figure 5-6 and Figure 5-7. The total cost of this strategy is roughly $4 million, which corresponds to 121% of the cost of the $3.3 million baseline.

Table 5-2: CubeSat Program Duration by Milestone (2:1 λ P 3)

Milestone	Kickoff	SRR	PDR	CDR	PSR	Delivery	Launch	Deploy	EOM
	Satellite 1								
Time until next (months)	2 mo.	5 mo.	6 mo.	11 mo.	-	-	-	-	-
	Build & test both s/c simultaneously					Modify s/c 2 based on s/c 1 testing			
	Time to s/c 1 completion: 24 mo.					Post-delivery time: N/A			
	Satellite 1 duration: 24 mo.								
	Satellite 2								
Time until next (months)	-	3 mo.	3 mo.		1 mo.	3 mo.	1 mo.	2 mo.	End
	Time to delivery: 7 mo.					Post-delivery time: 6 mo.			
	Satellite 2 duration: 13 mo.								
	Total program duration: 37 mo.								



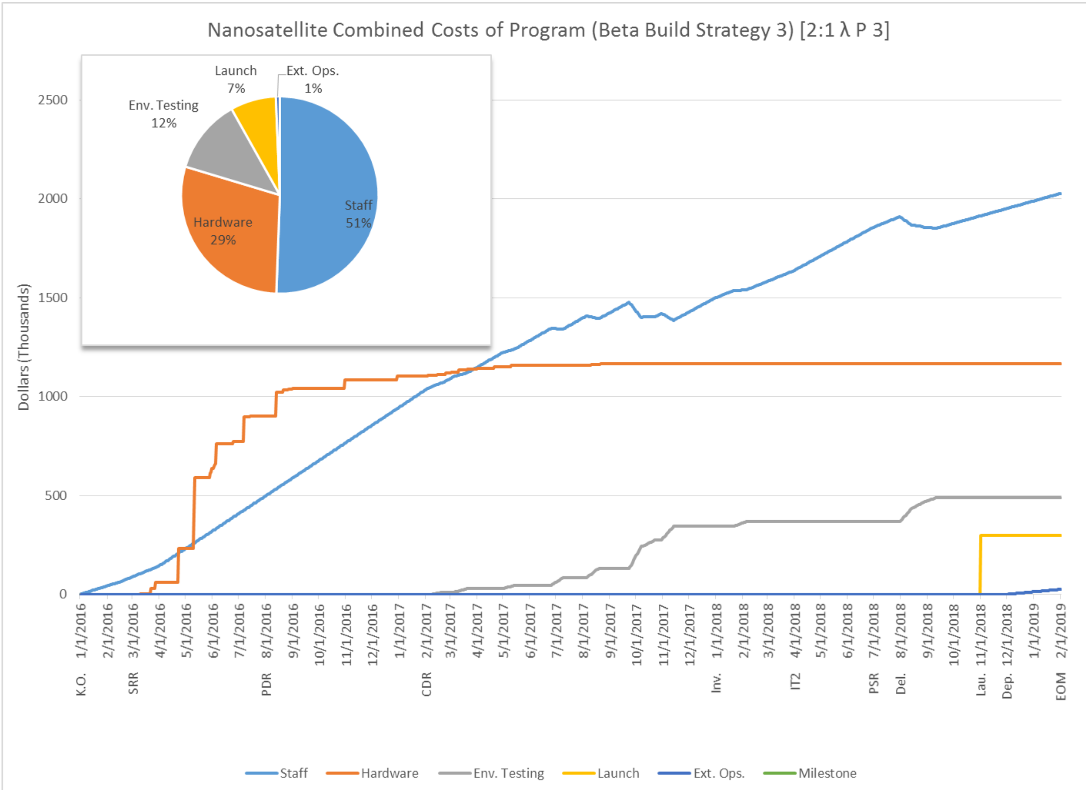


Figure 5-6: Nanosatellite Combined Costs of Program (2:1 λ P 3)

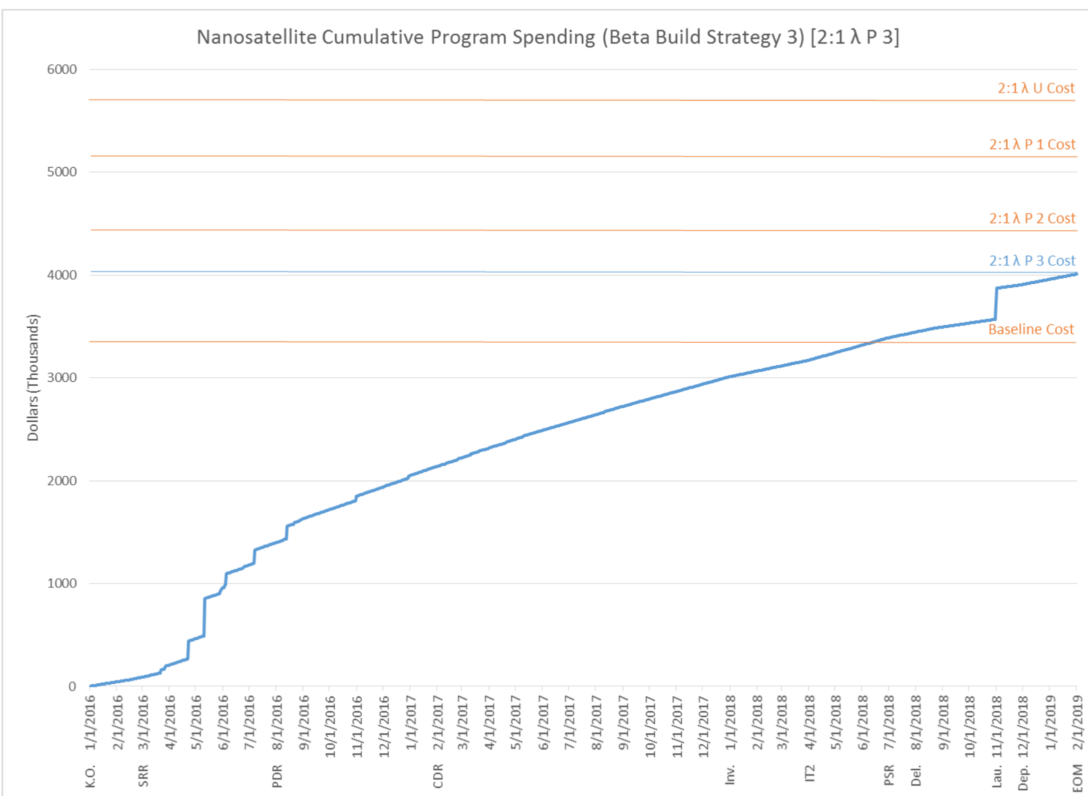


Figure 5-7: Nanosatellite Cumulative Program Spending (2:1 λ P 3)

5.5 Comparison and Discussion

The question which this analysis tries to answer is not, “how does a group spend the least amount of money to get their first CubeSat into space?” but rather, “how does a group spend the least amount of money to get a CubeSat into space that works?” With this in mind, of all the strategies analyzed here, we would most recommend the second beta build strategy (2:1 λ P 2). A lot can be learned from on-orbit operations which simply cannot be simulated on the ground. The time during mission operations of the first spacecraft should be utilized to investigate and diagnose the cause of any operational faults with the spacecraft and fix these issues on the second. One of the more challenging parts of spacecraft engineering is achieving flawless integrated operation, which is reflected in the operations of failed CubeSat missions. The cause of these failures was most often a “no contact failure’ – a failure of the successful operation of the integrated bus, which beta build strategy 2:1 λ P 2 validates (and provides the opportunity to fix, if necessary, for the second spacecraft). Then, by the time the team is ready to launch their second spacecraft, as long as the team is satisfied with the testing and operation of their payload on the ground, the team will be much more confident that their mission will be successful, while having saved a substantial amount of money by having built both spacecraft simultaneously and having not integrated the payload on the first spacecraft. It is difficult to quantify the added probability of success that utilizing one of these alternate build strategies would add, except by using the historical data to forecast future chances of success. Based on the data of CubeSat missions through the end of the year 2015, there is an 80% chance that the 1:1 λ baseline strategy (where, if it is a group’s first, one CubeSat is built and it is completely successful) will not work for a new CubeSat engineering team. So, rather than subject themselves to the much more likely scenario where they would end up having to then fix the faults from the first spacecraft on a second (2:1 λ U – 173% of the cost of the baseline), why not spend an additional 33% up-front (2:1 λ P 2) to gain the added confidence (and on-orbit validation) that the spacecraft will work by the time the payload is integrated (historically, 72% of a group’s second successfully launched CubeSat have met mission objectives, 30% more than those of a group’s first) [2] [1] [42]. The total program cost analysis results are summarized in Table 5-3.

Table 5-3: Summary of Total Program Cost Analysis Results

Strategy	Description	Total Program Cost	% Increase from 1:1 λ*	% Decrease from 2:1 λ U*
1:1 λ	3U CubeSat baseline. One s/c is built and is a total success.	$3.3 million	N/A	N/A
2:1 λ U	1:1 λ is taken, but the 1st s/c fails. The team then rebuilds 2nd s/c from scratch, but in shorter time b/c of experience.	$5.7 million	+73%	N/A
2:1 λ P 1	Plan to build 2 initially. The time after 1st launch is shortened b/c both s/c are built & tested simultaneously.	$5.1 million	+53%	-20%
2:1 λ P 2	Same as 2:1 λ P 1, but do not integrate a payload on the 1st s/c.	$4.4 million	+33%	-40%
2:1 λ P 3	Do not launch the 1st s/c at all – instead, lengthen testing on the ground. Still assumes 1st s/c is completely expendable.	$4.0 million	+21%	-52%

*In terms of 1:1 λ cost

Chapter 6: Conclusion

6.1 Summary of Results and Findings

CubeSats have opened up the space industry to a wider range of participants. By offering a standard form factor which is being accepted by an ever-increasing pool of service providers for secondary payloads, and anticipating dedicated launches, CubeSats have grown rapidly in their utilization for a variety of space applications. As CubeSats are typically much less expensive than their larger, more complex, cousins within the spacecraft industry, they are more accessible to entry-level teams with limited resources.

These limited resources often associated with CubeSat engineering groups, coupled with time-constraints levied by the launch service provider's expected delivery dates, often cause teams utilizing CubeSats to accept a higher amount of risk, purchasing fewer if any engineering model components and taking a more aggressive build and integration posture; accelerated and/or truncated testing is common. This shortened testing, specifically when it results in insufficient integrated testing, as well as an over-reliance on commercial off-the-shelf components and additional factors associated with their smaller size, have been driving factors in CubeSats' comparatively poorer operational performance: only one-fifth of successfully launched CubeSats thus far have met all of their mission objectives, with twice as many CubeSats failing to meet any of their mission objectives as compared with the rest of the satellite industry (by percentage of successfully launched spacecraft) [1] [2]. CubeSats are still in their infancy, however, with experience utilizing this platform greatly increasing probability of success: 72% of a group's second successfully launched CubeSats have achieved some or all of their mission objectives as opposed to only 42% for a group's first [2] [42].

This research discussed the testing and integration processes of CubeSats, providing examples from the Microwave Radiometer Technology Acceleration (MiRaTA) 3U CubeSat, in order to offer insights into how the practices behind CubeSat engineering can be improved, as well as to aid in increasing experience with the subject for entry-level groups. This research also analyzed CubeSat mission histories, with a particular emphasis on mission failures, to determine the most common sources of these failures so that they can be better mitigated in the future.

Leveraging this information, as well as data from three MIT Space Systems Laboratory and MIT Lincoln Laboratory CubeSat projects, three alternate build strategies, here referred to as "beta build strategies" were proposed which attempt to increase a CubeSat's probability of success while minimizing the natural associated increase in cost. The core money-saving principle of these alternate build strategies is that a team spends less money if it plans to build two flight iterations of its CubeSat from the beginning, as compared to the scenario in which a second is built only after an initial failure. To compare these build strategies to more traditional approaches, the total program costs were analyzed and compared to those of a baseline program and to a program which fails on its first attempt and succeeds on its second. The results of this analysis showed that planning to build two satellites from the beginning (Beta Build Strategy 1; 2:1 λ P 1) corresponds to roughly a 53% increase from the baseline, one-satellite build strategy (1:1 λ), and costs 20% less than if the group were to decide to build two satellites only after having their first fail (2:1 λ U). By varying additional parameters, such as deciding not to integrate the spacecraft's payload on the first spacecraft (Beta Build Strategy 2) or increasing testing on the ground rather than launching the first spacecraft (Beta Build Strategy 3), the cost decreases even further, with an additional 33% and 21% increase from the baseline, respectively. These strategies argue that making the decision to spend slightly more up-front is a compelling alternative to the historically poor-performing single-build approach, which ends up leading to spending more money in the long-run if the decision to build a second spacecraft arises only after a group's first spacecraft fails to meet its mission objectives.

6.2 Application

This research reflects on the history of the CubeSat industry thus far, and analyzes the testing and integration strategies of the MiRaTA project as a case study, in order to provide insights into the art of CubeSat engineering and how the strategies behind this practice may be improved. This research follows this analysis with the proposal of alternative options to the traditional build approach in an attempt to increase probability of CubeSat mission's successes while minimizing increased spending.

During the cost portion of this analysis, a conservative approach was taken with definitions and assumptions, favoring assumptions which would result in more, rather than less, spending. While analyzing the cost of the alternate build strategies, for example, it was assumed that the same amount of hardware (both engineering and flight) was purchased for the second spacecraft as the first. This was done intentionally, in order to establish the conservative, upper-limit value for what these scenarios would cost. In actual practice, a group may decide they do not need twice as many engineering model components in order to build two spacecraft, as the lessons learned from the EM components of the first apply equally to the second. It should also be noted that vendors often give discounts when a group orders multiples of the same component; these discounts, which could apply to the beta build strategies which purchase the hardware for two flight spacecraft simultaneously, are not factored into the analysis. In addition, groups may adjust their testing in such a way that differs from this analysis in order to further cut cost. Due to the many unique aspects of a CubeSat engineering group, the ideal approach for one group will not be identical to that of another, and groups must therefore conduct their own analysis into what the best approach would be for them. Certainly, specific cost values as well as assumptions made in analyzing cost will differ among groups. This research attempts to provide the framework around which that analysis can begin.

6.3 Future Work and Other Considerations

CubeSats are still relatively new in the space industry. With this research covering data current through the end of calendar year 2015, only 16 years of history was available to be analyzed. As CubeSats continue to be utilized and familiarity and experience with the platform continues to increase, so too will the number of their successful applications.

The increasing presence of organizations choosing to either utilize CubeSats for their own space missions, or to provide services to those who do, opens up the flexibility in design and build processes for CubeSat groups. Secondary payload opportunities have increased substantially to cater to this growing industry, with NASA's CubeSat Launch Initiative (CSLI) and Educational Launch of Nanosatellites (ELaNa) programs, which provide CubeSat engineering groups more opportunities to space than ever before. As dedicated CubeSat launch service providers begin to conduct operations within the next couple of years, CubeSat groups will have even more launch opportunities and flexibility in the design of their mission architectures. With commercial options for launch, CubeSat groups can gain more control of their schedule, picking a launch date at some point during their build process when they are ready to do so, rather than being assigned a launch date as a secondary payload. Since one of the major contributors to CubeSat mission failures has been insufficient integrated testing, which is exacerbated by late component delivery dates that compress the time period allocated to testing, groups may decide that getting to choose their own launch date is worth paying the launch cost themselves. By choosing a dedicated launch provider, CubeSat groups could start their design and build process and wait to choose their launch date once they have a better idea about how much time they need to finish. The cost-benefit analysis could shift even more in-favor of a group paying for their own launch when considering the cost difference between the cost of a launch and the cost of completely rebuilding their satellite if the first failed because they had to rush in order to meet their assigned secondary payload launch date.

This research can be extended to cover other areas of interest within CubeSat engineering. As the CubeSat industry matures, more groups will gain experience with applying CubeSats to a variety of space missions. Many university groups are now on their second or third CubeSat build, with commercial groups continuing to produce further iterations of their CubeSats. It would be interesting to extend this analysis to cover these multiple-iteration build programs, attempting to quantify the relative cost increase between each iteration. This would be especially interesting as more groups apply CubeSats to constellation mission architectures, in which they could build ten or more of the same spacecraft. In such a build process, groups would no longer require the same amount of testing on their tenth iteration as their first, nor would they need to take as much of a risk-adverse posture in the design of the build timeline for each spacecraft if they were planning to build so many from the beginning. Such aspects clearly enable cost-savings with each continued iteration (as is likely very well-known for commercial groups already capitalizing on this fact). It would be interesting to quantify this within a framework similar to this analysis; university groups wishing to utilize CubeSat builds primarily for the educational opportunities it gives their student engineers could benefit greatly from such a long-term strategy, as the per-unit cost continues to decrease with each successive build.

As the number of applications for CubeSats continue to grow, it would also be interesting to apply this analysis to compare programs by field. The build strategies and risks of programs applying CubeSats to atmospheric sensors likely differ greatly from those applying CubeSats to the field of multispectral imagers, for example. These differences could be even greater when considering new applications for CubeSats as well as technology demonstration missions. Integrating new technologies or applying the same technologies in new ways certainly carries with it more risk; it would be interesting to extend this research to analyze the relative merits of alternative build strategies based on the differences of CubeSats by field.

It is this author's hope that the CubeSat industry continues to grow as it has, with more groups continuing to embrace the CubeSat platform for their space mission applications. The author is very grateful for the opportunity to work with CubeSats as they provided the framework in which the practice of engineering satellites could be learned. Finally, it is the author's hope that the MIT Space Systems Laboratory and Lincoln Laboratory continue to work together on CubeSat projects in the future.

References

[1] G. D. Krebs, "Gunter's Space Page - CubeSat," 29 June 2015. [Online]. Available: http://space.skyrocket.de/doc_sat/cubesat.htm. [Accessed 29 June 2015].

[2] M. Swartwout, "CubeSat Database," St. Louis University, 29 June 2015. [Online]. Available: https://sites.google.com/a/slu.edu/swartwout/home/cubesat-database. [Accessed 29 June 2015].

[3] J. Puig-Suari, C. Turner and W. Ahlgren, "Development of the Standard CubeSat Deployer and a CubeSat Class PicoSatellite," *Proc. 2001 Aerospace Conference,* vol. 1, pp. 347-353, 2001.

[4] NASA, "About the Space Station: Facts and Figures," 10 April 2016. [Online]. Available: https://www.nasa.gov/mission_pages/station/main/onthestation/facts_and_figures.html. [Accessed 9 May 2016].

[5] Z. Manchester, "KickSat -- Your personal spacecraft in space!," 29 May 2015. [Online]. Available: https://www.kickstarter.com/projects/zacinaction/kicksat-your-personal-spacecraft-in-space/description. [Accessed 9 May 2016].

[6] S. Gao, K. Clark, M. Unwin, J. Zackrisson, W. A. Shiroma, J. M. Akagi, K. Maynard, P. Garner, L. Boccia, G. Amendola, G. Massa, C. Underwood, M. Brenchley, M. Pointer and M. N. Sweeting, "Antennas for Modern Small Satellites," *IEEE Antennas and Propagation Magazine,* vol. 51, no. 4, pp. 40-56, August 2009.

[7] I. Nason, J. Puig-Suari and R. Twiggs, "Development of a Family of Picosatellite Deployers Based on the CubeSat Standard," *Proc. 2002 IEEE Aerospace Conference,* vol. 1, pp. 457-464, 2002.

[8] A. Scholz and J.-N. Juang, "Toward open source CubeSat design," *Acta Astronautica,* vol. 115, pp. 384-392, 2015.

[9] NanoRacks LLC, "NanoRacks: The Operating System of Space," 9 May 2016. [Online]. Available: http://nanoracks.com/. [Accessed 9 May 2016].

[10] Innovative Solutions In Space, "3-Unit CubeSat Deployer," 9 May 2016. [Online]. Available: http://www.isispace.nl/brochures/ISIS_ISIPOD_Brochure_v.7.11.pdf. [Accessed 9 May 2016].

[11] K. Woellert, P. Ehrenfreund, A. Ricco and H. Hertzfeld, "CubeSats: Cost-effective science and technology platforms for emerging and developing nations," *Advances in Space Research,* vol. 47, pp. 663-684, 2011.

[12] Firefly Space Systems, "Firefly Alpha," 23 February 2016. [Online]. Available: 2http://www.fireflyspace.com/vehicles/firefly-a. [Accessed 23 February 2016].

[13] Firefly Space Systems, "Our News," 14 October 2015. [Online]. Available: www.fireflyspace.com/news/ournews. [Accessed 2 March 2016].

[14] Rocket Lab, "Electron Booking," 23 February 2016. [Online]. Available: http://book.rocketlabusa.com/. [Accessed 23 February 2016].

[15] Rocket Lab, "Rocket Lab News," December 2015. [Online]. Available: www.rocketlabusa.com/category/rocket-lab-news/. [Accessed 2 March 2016].

[16] Virgin Galactic, "Satellite Launch," 23 February 2016. [Online]. Available: http://www.virgingalactic.com/satellite-launch/. [Accessed 23 February 2016].

[17] Virgin Galactic, "LauncherOne Performance," 2 March 2016. [Online]. Available: www.virgingalactic.com/satellite-launch/l1-performance/. [Accessed 2 March 2016].

[18] Virgin Galactic, "LauncherOne Operation," 2 March 2016. [Online]. Available: www.virgingalactic.com/satellite-launch/l1-operations/. [Accessed 2 March 2016].

[19] J. Cutler, G. Hutchins and R. Twiggs, "OPAL: Smaller, Simpler, and Just Plain Luckier," in *14th AIAA/USU Conference on Small Satellites*, Logan, 2000.

[20] Planet Labs Inc., "About," 2016. [Online]. Available: https://www.planet.com/about.

[21] W. Blackwell, G. Allen, C. Galbraith, T. Hancock, R. Leslie, I. Osaretin, L. Retherford, M. Scarito, C. Semisch, M. Shields, M. Silver, K. Toher, K. Wight, D. Miller, K. Cahoy and N. Erickson, "Nanosatellites for Earth Environmental Monitoring: The MicroMAS Project," *AIAA/USU Conference on Small Satellites, Logan, UT,* 2012.

[22] W. Blackwell and K. Cahoy, *The Microwave Radiometer Technology Acceleration (MiRaTA) CubeSat,* Cambridge: Massachusetts Institute of Technology Lincoln Laboratory, 2013.

[23] W. Blackwell, G. Allen, K. Cahoy, A. Marinan and N. Erickson, "MicroMAS: A First Step Towards a Nanosatellite Constellation for Global Storm Observation," *27th Annual AIAA/USU Conference on Small Satellites, Logan, UT,* 2013.

[24] W. Blackwell and S. Braun, "TROPICS," MIT Lincoln Laboratory, 10 March 2016. [Online]. Available: https://tropics.ll.mit.edu/CMS/tropics/. [Accessed 9 May 2016].

[25] A. Marinan and K. Cahoy, *From CubeSats to Constellations: Systems Design and Performance Analysis,* Cambridge: Massachusetts Institute of Technology, 2013.

[26] J. Byrne and K. Cahoy, *Resource-constrained Avionics Design for CubeSats,* Cambridge: Massachusetts Institute of Technology, 2015.

[27] W. Blackwell, *Micro-size Microwave Atmospheric Satellite-2b (MicroMAS-2b) CubeSat,* Cambridge: Massachusetts Institute of Technology Lincoln Laboratory, 2015.

[28] W. Blackwell, R. Bishop, K. Cahoy, B. Cohen, M. DiLiberto and R. Leslie, "Radiometer Calibration Using Colocated GPS Radio Occultation Measurements," *IEEE Transactions on Geoscience and Remote Sensing,* vol. 52, no. 10, pp. 6423-6433, 2014.

[29] P. Dave`, *Using a co-located GNSS radio occultation payload for microwave radiometer calibration,* Cambridge: Massachusetts Institute of Technology, 2014.

[30] W. Marlow, K. Riesing, T. Nguyen, K. Cahoy, Z. Decker, E. Thompson and J. Byrne, *Attitude Determination and Control Approach to Achieve Co-Located Microwave Radiometer and GPS Radio Occultation Measurments on a Nanosatellite,* Breckenridge: AAS GNC Conference, 2015.

[31] T. Cordeiro, *Structural Design of the Microwave Radiometer Technology Acceleration (MiRaTA) CubeSat,* Lexington: Massachusetts Institute of Technology Lincoln Laboratory, 2015.

[32] The Aerospace Corporation, "Test Requirements for Launch, Upper-Stage, and Space Vehicles - SMC-TR-06-11 AEROSPACE REPORT NO. TR-2004(8583)-1 REV. A," El Segundo, 2006.

[33] R. Dybdal, "Communication Satellite Antenna Testing," *Instrumentation & Measurement Magazine, IEEE,* vol. 18, no. 3, pp. 28-33, 2015.

[34] K. Lee, "What is Space Radiation?," NASA, 5 March 2014. [Online]. Available: http://srag-nt.jsc.nasa.gov/spaceradiation/what/what.cfm. [Accessed 27 8 2015].

[35] J. Byrne, E. Thompson and R. Kingsbury, "Space Radiation Environmental Analysis of CubeSat Components," Massachusetts Institute of Technology, Cambridge, 2015.

[36] R. Aniceto, W. Lohmeyer and K. Cahoy, "Total Ionizing Dose Requirements for Low Earth Orbit Small Satellites," *Journal of Small Satellites (currently under review),* 2016.

[37] C. Lafleur, "Spacecraft Enclyclopedia," 2015. [Online]. Available: claudelafleur.qc.ca/Spacecrafts-index.html. [Accessed 08 June 2015].

[38] G. Dubos, C. Jean-Francois and J. Saleh, "Statistical reliability analysis of satellites by mass category: Does spacecraft size matter?," *Acta Astronautica,* vol. 67, pp. 584-595, 2010.

[39] J. Guo, L. Monas and E. Gill, "Statistical analysis and modelling of small satellite reliability," *Acta Astronautica,* vol. 98, pp. 97-110, 2014.

[40] M. Swartwout, "The First One Hundred CubeSats: A Statistical Look," *Journal of Small Satellites,* vol. 2, no. 2, pp. 213-233, 2013.

[41] A. Hein, Z. Lee and K. Cahoy, "MicroMAS Mission Analysis: Solar Array Deployment," *Jornal of Small Satellites (currently under review),* 2016.

[42] M. Swartwout, "CubeSats: Toys, Tools or Debris Cloud?," in *Gateway to Space Conference*, St. Louis, 2014.

[43] Spaceflight Incorporated, "Schedule & Pricing," 2015. [Online]. Available: http://www.spaceflight.com/schedule-pricing/. [Accessed 5 February 2016].

[44] C. Lowe and M. Macdonald, "Rapid model-based inter-disciplinary design of a CubeSat mission," *Acta Astronautica,* vol. 105, pp. 321-332, 2014.

[45] NASA, "NASA Awards First CubeSat-Class Launch Services Contract," 30 September 2013. [Online]. Available: http://www.nasa.gov/centers/kennedy/news/releases/2013/release-20130930.html. [Accessed 5 February 2016].