# Mission Concept User Guide

## University Nanosatellite Program

Initial Release, May 2023

**Visit:** universitynanosat.org
**Contact:** info@universitynanosat.org

Approved for public release; distribution is unlimited. Public Affairs release approval AFRL-2023-1750
EXTDOC-315 Rev. A

| This document owned and controlled by AFRL/RV: University Nanosatellite Program | | | |
|---|---|---|---|
| REV | Date | Reason | Personnel |
| -- | 5/22/2023 | Initial Release | UNP |
| A | 3/28/2024 | Updated Events to become generic timeframe | UNP |

# Acronym List

| | |
|---|---|
| **ADC or ADCS** | Attitude Determination & Control Subsystem |
| **AES** | Advanced Encryption Standard |
| **AFOSR** | Air Force Office of Scientific Research |
| **AF or USAF** | United States Air Force |
| **AFRL** | Air Force Research Laboratory |
| **AFRL/RV** | Space Vehicles Directorate |
| **AI&T** | Assembly, Integration, and Test |
| **CDH or C&DH** | Command and Data Handling |
| **CAD** | Computer-Aided Design |
| **CBC** | Cypher Block Chaining |
| **CDR** | Critical Design Review |
| **CE** | Chief Engineer |
| **CET** | Command Execution Test |
| **CG** | Center of Gravity |
| **CG/MOI** | Center of Gravity/Moment of Inertia |
| **CM** | Configuration Management |
| **COM** | Communications Subsystem |
| **CONOPS** | Concept of Operations |
| **COTS** | Commercial Off the Shelf |
| **CSLI** | CubeSat Launch Initiative |
| **CVCM** | Collectable Volatile Condensable Material |
| **DitL** | Day in the Life |
| **DoD** | Department of Defense |
| **EAT** | Expert Area Telecon |
| **ECB** | Electronic Code Book |
| **EGSE** | Electrical Ground Support Equipment |
| **EIDP** | End Item Data Package |
| **ELaNa** | Education Launch of Nanosatellites |
| **EM** | Engineering Model |
| **EMI/EMC** | Electromagnetic Interference and Electromagnetic Compatibility |
| **EPS** | Electrical Power Subsystem |
| **ER&T** | Exploration Research and Technology Program |
| **ESD** | Electro-static discharge |
| **FAA** | Federal Aviation Administration |
| **FCC** | Federal Communications Commission |
| **FFT** | Full Functional Test |
| **FM** | Flight Model |
| **FSR** | Flight Selection Review |
| **GCM** | Galois/Counter Mode |
| **GSE** | Ground Support Equipment |
| **IARU** | International Amateur Radio Union |
| **IBF** | Install Before Flight |
| **ICD** | Interface Control Document |
| **ITAR** | International Traffic in Arms Regulations |
| **LEO** | Low Earth Orbit |
| **LV** | Launch Vehicle |
| **MEL** | Master Equipment List |
| **MGSE** | Mechanical Ground Support Equipment |
| **MO** | Mission Objective |
| **MPPT** | Maximum Power Point Tracking |
| **MSC** | Mission Success Criteria |
| **NASA** | National Aeronautics and Space Administration |
| **NOAA** | National Oceanic and Atmospheric Administration |
| **PCB** | Printed Circuit Board |
| **PDR** | Preliminary Design Review |
| **PI** | Principal Investigator |
| **PIR** | Pre-Integration Review |
| **PM** | Program Manager |
| **PMR** | Program Management Review |
| **PSR** | Pre-Ship Review |
| **PSR-LV** | Pre-Ship Review to the Launch Vehicle |
| **QA** | Quality Assurance |
| **RBF** | Remove Before Flight |
| **RF** | Radio Frequency |
| **RVM** | Requirements Verification Matrix |
| **SCT** | Simulated Communications Test |
| **SCR** | System Concept Review |
| **SERB** | Space Experiments Review Board |
| **SRR** | System Requirements Review |
| **SSP** | Small Satellite Portfolio |
| **STEM** | Science, Technology, Engineering, & Mathematics |
| **STP** | Space Test Program |
| **SV** | Space Vehicle |
| **TML** | Total Mass Loss |
| **TNC** | Telemetry & Command |
| **TRL** | Technology Readiness Level |
| **UHF** | Ultra-High Frequency |
| **UNP** | University Nanosatellite Program |
| **VC** | Visibly Clean |

| Link/Name | Notes |
|---|---|
| Universitynanosat.org | UNP Website |
| nasa.gov/directorates/heo/home/CubeSats_initiative | CSLI Website |
| smallsat.org | Small Satellite Conference Website |
| outgassing.nasa.gov | Provides outgassing information for many materials |
| software.nasa.gov/software/MSC-26690-1 | NASA Debris Assessment Software (DAS) Download |
| ntrs.nasa.gov/api/citations/20205002380/downloads/NASA%20TP-2020-5002380_DAS_UsersGuide_final.pdf | NASA Debris Assessment Software (DAS) User Guide |
| nepp.nasa.gov/npsl | NASA Parts Selection List |
| maptis.nasa.gov | NASA Materials and Processes Technical Information System |
| standards.nasa.gov/standard/NASA/NASA-STD-5003 | NASA Fracture Control Requirements |
| standards.nasa.gov/standard/msfc/msfc-std-3029 | NASA Guideline for selection of Metallic Materials |
| Space Mission Engineering, The New SMAD (Text) | Space mission engineering overview and guidance in a widely regarded textbook |

## Table of Contents

# 1 Introduction

## 1.1 User's Guide Introduction

This guide is specific to the University Nanosatellite Program (UNP) Mission Concept Program. It focuses heavily on mission design and initial systems engineering processes and only briefly covers detailed design, assembly, integration, and testing. Information on those topics can be found in the UNP Nanosatellite User Guide and other associated UNP resources. It also assumes the reader has entry-level familiarity with engineering, space, and satellite concepts such as mechanical and electrical calculations and terms, orbits, and CubeSats. UNP provides much of this information during the Kickoff and Mission Design presentations.

All publicly released UNP resources are available on the resources tab of the UNP website: universitynanosat.org. Whether you are a new team embarking in UNP for the first time, an unaffiliated team using our resources, or one that has been in the program for years, this document is designed to help you. We expect you'll find the included guidance beneficial while you develop your missions.

The University Nanosatellite Program was founded in 1999 and has provided guidance and support to over 100 small satellite missions across all stages of small spacecraft development and operations. This wealth of experience and the lessons learned have informed all our current resources, and we believe they can provide benefit to any satellite development team, from beginner to professional.

There are many approaches to small spacecraft design, as well as many approaches to systems engineering. Each has their place, and what works for one team may not work for another. The UNP approach is not meant to be the one and only approach to building small satellites, but rather an approach that we believe best trains students in developing and flying successful small satellites with meaningful missions in a resource constrained, educational setting.

From the Principal Investigator (PI) to the newest team member, there is something for everyone in this guide. While parts of it will be more relevant at some times than others, we recommend that everyone on the team read this document and be familiar with its contents for the entire length of the program.

This document covers most of what you'll need to know during your time in UNP. Chapter 1 introduces the program and provides university expectations. Chapter 2 covers the satellite development process, providing general space industry knowledge and UNP's approach. Chapter 3 overviews the Mission Concept Program. Chapter 4 explains the deliverables required throughout the program. Chapter 5 provides design guidelines UNP levies on teams building satellites. These guidelines are provided for reference, and many will not be applicable to the Mission Concept Program

In this document, we've tried to incorporate many lessons learned. However, each program is unique, and new challenges will always arise. Should you find yourself in a situation that is not covered, please contact us, and ask questions. Between us, this guide, and the larger satellite community we have access to, we should have you covered.

## 1.2 The University Nanosatellite Program

UNP is a Department of the Air Force – Air Force and Space Force – STEM (Science, Technology, Engineering, and Mathematics) initiative. Managed by the Air Force Research Laboratory Space Vehicles Directorate, it is a partnership with AFRL's Air Force Office of Scientific Research (AFOSR), the DoD Space Test Program (STP), the National Aeronautics and Space Administration's CubeSat Launch Initiative (NASA CSLI) and Exploration Research and Technology Program (NASA ER&T), the Missile Defense Agency (MDA), and The Space Warfighting Analysis Center (SWAC). UNP is a program within the AFRL Small Satellite Portfolio (SSP) headquartered at Kirtland Air Force Base in Albuquerque, New Mexico. Sitting within SSP allows UNP, and our university teams, to draw from the expertise of many AFRL engineers.

These organizations serve various roles within UNP. AFOSR, NASA, MDA, and SWAC, provide the funding for the universities, STP provides the launch, and UNP (at AFRL/RV) provides the programmatic structure and technical support. In this Mission Concept Program, NASA CSLI and ER&T also play significant roles in the programmatic structure and curriculum. The group you will interface with the most is UNP.

UNP manages a variety of different programs. The traditional UNP programs are known as Nanosatellite Programs (or cycles) and involve the complete satellite lifecycle from concept through operations. These are multi-year programs with various reviews, down selects, and funding opportunities. Similar to the Nanosatellite Programs are Technology Insertion Programs. These typically involve the complete satellite development process, but the mission is more closely tied to a U.S. Department of Defense (DoD) program. In these programs, hardware, experiment plans, and other directives may be provided to the school rather than expecting the school to develop them. They also put a slightly lower emphasis on education of students, with a higher emphasis on technology development. Finally, UNP offers multi-month mission design programs focused on education in the early phases of systems engineering. These are called Mission Concept Programs and are the focus of this User Guide.

### 1.2.1 Program Objectives

UNP has three program objectives, illustrated in Figure 1-1. The primary objective of UNP is education. As a STEM program, UNP's purpose is to educate students to be better satellite systems engineers. This is the foundation of the program and underpins every decision made by UNP. This is why UNP is a "cradle to grave" program, allowing you to experience a mission from its inception through its end-of-life. UNP strives to emphasize lessons learned from building complex systems.

The secondary objective of UNP is university lab development. Some universities have vast resources, enabling them to continually develop satellites, but many do not. UNP strives to help developing labs reach their full potential.

The third objective of UNP is technology development at the component, system, and architecture levels. Universities are the birthplace of innovation, and UNP aims to capitalize on that innovation for the DoD's benefit. Therefore, military relevance is an important and key part of UNP. Sometimes NASA or other stakeholders will partner with UNP on programs. In these cases, relevance to NASA is equally important as relevance to the DoD.

Figure 1-1 Objectives of the University Nanosatellite Program

## 1.3 NASA Partnerships

Multiple NASA groups, CSLI and ER&T, are primary partners in the Mission Concept Program. Based at Kennedy Space Center (KSC), both provide funding and guidance to educational space programs in the US.

### 1.3.1 CubeSat Launch Initiative (CSLI)

The CubeSat Launch Initiative (CSLI) is a Space Operations Mission Directorate (SOMD) initiative in partnership with the Science Mission Directorate (SMD), Space Technology Mission Directorate (STMD), Office of STEM (OSTEM) and the Launch Services Program (LSP). The initiative was formally established in 2010 with the first Announcement of Opportunity to provide access to space for small satellites (CubeSats) developed by U.S. educational institutions, nonprofits with an education/outreach component, and NASA Centers and programs for workforce development. It provides launch opportunities to a variety of U.S. CubeSat developers who build small satellite payloads that fly as auxiliary payloads on previously planned launches or commercial missions to low Earth orbit and deep space destinations as well as International Space Station deployments.

The initiative creates a unique and mutually beneficial partnership between NASA and the organizations selected to participate. Partnering organizations are provided with a mechanism to conduct scientific research in the space environment and advance the development of various technologies. It enables students, educators, and faculty to obtain hands-on flight hardware development experience and the efforts are a cornerstone in the development of cutting-edge NASA enabling technologies like laser communications, next generation avionics approaches, power generation, distributive sensor systems, satellite-to-satellite communications, and autonomous movement. Leveraging these missions for collaboration optimizes NASA's technology investments, fosters open innovation, and facilitates technology infusion, ensuring the greatest national benefit.
CSLI provides up to $300,000 in funding for integration and launch services for any selected CubeSat, regardless of size. This is typically enough to launch a 3U CubeSat into LEO.

NASA's CubeSat Launch Initiative is intended to expand U.S. interest in Science, Technology, Engineering, and Mathematics (STEM). With the focus on education, the projects selected have a primary focus area of education or workforce development. Secondary focus areas are Scientific Research and/or Technology Development or Demonstration. CubeSats are playing an increasingly larger role in the agency's exploration, science, technology, and educational investigations and therefore assisting NASA in meeting its Strategic, Science, and STEM goals. The initiative encourages participation by Minority Serving Institutions (MSIs) and values multi-university collaborations and mentoring that will ultimately extend to a more diverse workforce.

Visit the CSLI website to learn more about the initiative, and see previously selected projects, CubeSats, and Launches – known as Educational Launch of Nanosatellite (ELaNa) missions. CSLI is a function of the Space Operations Mission Directorate (SOMD) Launch Services Office, which is located at NASA Headquarters in Washington, D.C. and is managed by the Launch Services Program located at Kennedy Space Center, Florida.

### 1.3.2 Exploration Research and Technology Programs (ER&T)

ER&T was created to assist universities in understanding Kennedy Space Center's research and technology portfolio, core capabilities, and research opportunities in alignment with NASA's Small Spacecraft Strategic Plan. ER&T strives to increase collaboration with university principal investigators and leverage KSC's extensive capabilities to support university small satellite activities.

## 1.4 Points of Contact and Further Information

| University Nanosatellite Program | CubeSat Launch Initiative | Exploration Research and Technology Programs |
|---|---|---|
| **Emi Colman**<br>*UNP Deputy Program Manager*<br>info@universitynanosat.org<br>505-846-6765 | **Norman L. Phelps**<br>*NASA Launch Services Program Mission Manager*<br>norman.l.phelps@nasa.gov | **Jose Núñez**<br>*NASA KSC University Partnerships/Small Satellite Capabilities Manager*<br>jose.l.nunez@nasa.gov |
| **Beth Hemmerich**<br>*UNP Program Coordinator*<br>info@universitynanosat.org<br>505-846-6765 | **Liam Cheney**<br>*NASA Launch Services Program Mission Manager*<br>liam.j.cheney@nasa.gov | |
| universitynanosat.org | nasa.gov/directorates/heo/home/cubesats_initiative | |
| For more information on AFRL, visit:<br>afresearchlab.com | CSLI is a function of the Space Operations Mission Directorate (SOMD) Launch Services Office, which is located at NASA Headquarters in Washington, D.C. and is managed by the Launch Services Program located at Kennedy Space Center, Florida. E-mail questions to: hq-launchservices@mail.nasa.gov | |

## 1.5 UNP Expectations for the University Team

The following chapters of this document will explain in detail UNP's expectations of each team. Note that much of this material applies to multi-year satellite projects more so than multi-month summer projects. Since most teams participating in the Mission Concept Program hope to develop their concepts into real satellites, this material is included. Developing good team habits, practices, and culture will set you up for success as the project increases in complexity.

### 1.5.1 Student Involvement

As UNP is a STEM program, the most important expectation is that each university program is deeply rooted in student involvement. Except for the PI, it should be a program run by students. We require one student assigned to lead the team and interface with UNP – the student Program Manager (PM). In addition, it is expected that the Chief Engineer (CE) of each team be a student. The CE works with the PM to guide the team in satellite development.

This isn't to say that professional engineers, university professors, and members of industry or government can't assist or provide support. This will be the first time many students have participated in satellite design and the support of professionals and academics is not just helpful, it's necessary. The keyword here, though, is support.

It should be the students who are making design decisions, the students who are deciding what algorithms they are using, and the students who test and operate the satellite. Ultimately, it will be the students who are benefitting from the educational experience of developing, building, and flying a satellite.

While some students will be interning with UNP for a portion of the program, the effort should not be limited to them. UNP will ensure that remote students at the university can participate as fully as possible, and the interning students should strive for this as well. The interning students will likely be the leaders of their team due to increased feedback and interaction with UNP, but they should approach the internship with the mindset that their team fully includes the personnel back at the university.

Case Study 1-1 Lack of Student Involvement

| The Case of Over Involved Industry | |
|---|---|
| **Situation:** | A UNP PI had a close relationship with a local branch of a company developing/selling CubeSat hardware. The company had several pieces of CubeSat hardware that they wanted to test in space, so they made a deal with the PI to donate the hardware to the team. |
| **Problem:** | The company donated the entirety of the satellite bus, and the PI dictated that this is what the students would use. The students were not part of this decision. Trade studies were ignored, as was requirement flow down. It was a decision made by non-students, eliminating student education and much of the systems engineering.<br>To be clear, the problem was not donated hardware. Universities often receive hardware donations or heritage systems that constrain their mission. The problem was that every piece of hardware was dictated to the student team, removing the students from the design process. |
| **Solution:** | Systems engineering education is a primary goal of UNP. UNP raised concern about the lack of student involvement at every review, which the PI and PM never addressed. Since student involvement is critical to UNP, they did not progress past their down select review. |

### 1.5.2 Team Structure

Every university must have a student team. There are many possible team organization methods. UNP attempts to foster innovation in our universities, even with team organization. However, certain structures, as seen in Figure 1-2, prove most effective for most teams. Ultimately it doesn't matter what specific names are given to roles or how responsibilities are divided if all are captured and communicated.

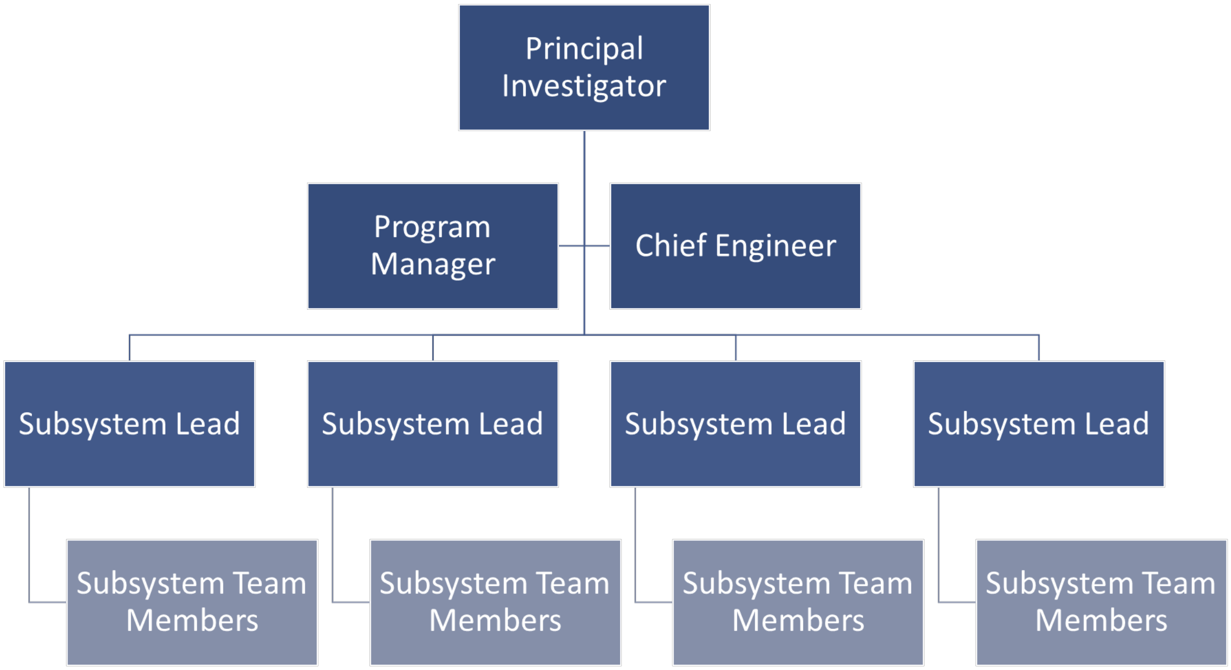


Figure 1-2 Example Team Structure

#### 1.5.2.1 Principal Investigator

The PI is a required role, held by a professor at the university. The PI is usually the one who conceived the mission concept and creates the vision for the students to follow. The PI is also expected to do the following:

- Provide technical and programmatic guidance to the students and/or help find other experts if unable.
- Oversee the lab and student team.
- Serve as the liaison between the team and the university, navigating complexities of university administration.

The PI also serves as a constant throughout longer programs, not being subjected to the high levels of student turnover seen in multi-year projects. The PI provides consistency, guidance, and vision to the students – especially the student leadership. There can be more than one PI, and a co-PI can be a non-faculty member; however, there must be at least one faculty PI.

#### 1.5.2.2 Program Manager

The team PM is a required role that must be held by a student at the university. It is the PM's job to translate the PI's vision into a schedule and concrete tasks. The PM is responsible for organizing and managing the student team members, keeping subsystem teams appropriately staffed, and ensuring smooth transitions of knowledge and leadership.

Part of the PM's responsibility is to keep the students on the team motivated. Keeping momentum high can be challenging, especially with semester breaks. The PM must be a motivator who keeps students focused and encouraged, but also recognize when a student is stretched beyond their means. While they are an encourager, they should still be a realist. Schedules must be built on reality rather than optimism.

Above all, the PM's responsibility is to ensure the students are working towards a common goal. The PM is the team's advocate to the PI and UNP and strives to balance the pressure of higher authority against the reality of the team's capabilities.

| The Case of Imbalanced Teams | |
|---|---|
| **Situation:** | The PM of a team was updating the personnel list with newly accepted subsystem team members. They noticed that while most teams had approximately three members, the Flight Software (FSW) team only had two members, and the Attitude Determination and Control (ADC) team had six members. |
| **Problem:** | While subsystem team imbalances aren't inherently bad, this FSW team was falling behind, unable to progress at the same rate as other teams. |
| **Solution:** | Ideally, one or two members could be moved from ADC to FSW, but the PM recognized that skillsets between teams may not be compatible. FSW team members needed to understand coding in C flight computer interfaces, while ADC members may be more mechanically focused. Fortunately, the PM maintained lists of each team member's skills, including programming languages. One ADC team member was fluent in C. After discussion with the student and both team leads, the student was transferred to the FSW team. |

### 1.5.2.3 Chief Engineer

The team CE is a required role that must be held by a student at the university. The CE is the team's lead systems engineer, ensuring the requirements of the mission are being met. The Requirements Verification Matrix (RVM) discussed in 4.2.7 of this document is the CE's primary tool.

Since it is the CE's responsibility to make sure all requirements are met, they are also responsible for final approval of technical decisions and trades. It's also the CE's responsibility to ensure that all satellite budgets (link, mass, data, etc.) are balanced.

The CE is also responsible for all system-level design and testing. "System level" means all designs and tests that incorporate more than one subsystem. Also, all RVM verifications—whether test, inspection, or analyses—fall under the CE's responsibility for completion and approval.

Ultimately the CE's job is to ensure that the team adheres to sound engineering — both in design and practices. The CE doesn't need to be an expert in every subsystem, but they need to apply engineering judgment to discern good design from bad.

**Note**

The statement that the CE ensures every component trade and design decision meets the satellite requirements does not mean the CE must write every trade or make every decision. The CE is the approving authority for such documents. They must ensure any design decision meets the requirements and needs of the satellite.

Subsystem leads are experts in, and advocates for, their subsystem meaning they don't always hold a system perspective. A structural engineer may want 40% of the mass budget allocation to make the structure as robust as possible, but the CE knows that the mass budget must be split equitably between subsystems. It is the CE's job to ensure the subsystems meet their requirements and work well within the system.

### 1.5.2.4 Common Leadership Challenges

The most common challenge is balancing the leadership of the PI, PM, and CE. There will be overlap between the roles and responsibilities may be shared across positions. The PI, PM, and CE should be in sync and lead the team together. If one has too much control, diminishing the roles of the others, responsibilities and tasks often slip through the cracks. The key is to ensure all tasks are being done and that the leadership is working together as a team.

The trickiest part of this leadership structure is navigating the relationship between the PI and the student leaders. The PI is a professor and therefore has substantial authority and responsibility. It can be difficult for the student leaders to push back confidently and respectfully. While the PI is the ultimate responsible party for the program, the PI should take the concerns and perspectives voiced by the student leaders seriously.

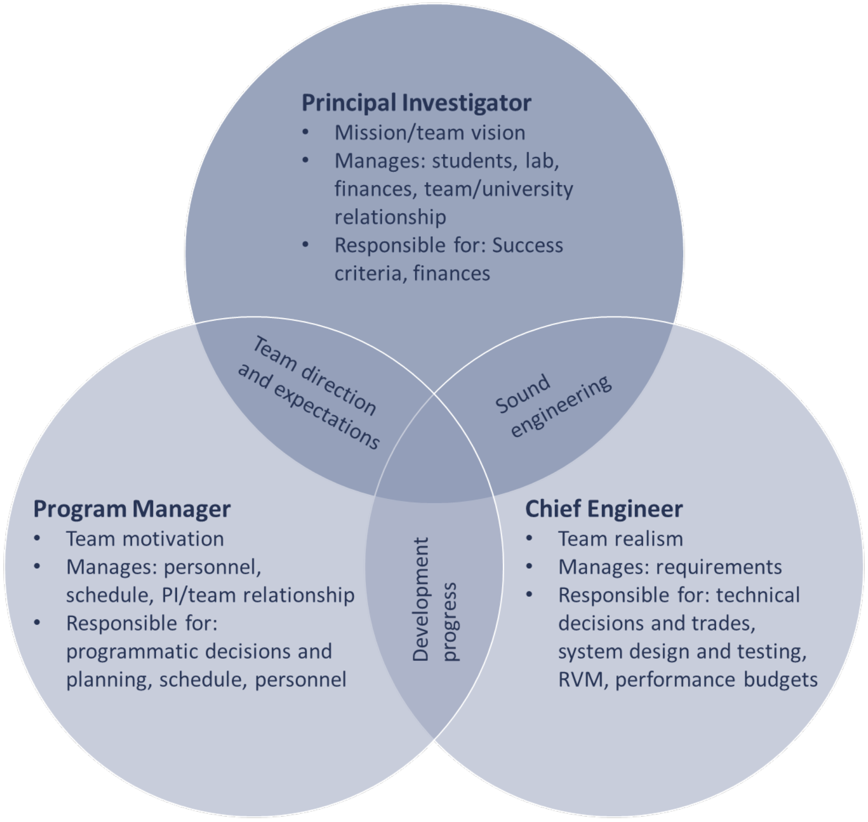


Figure 1-3 Team Leadership

**Note**

The different perspectives of the PI, PM, and CE create inherent relationship challenges.

The PI is commonly the team optimist, looking at the team from a higher level. The PI believes the students are more capable than they know and attempts to inspire them. The CE may be more pessimistic, struggling through the challenges of engineering a satellite. They may see the deadlines set by the PI and PM as impossible due to the overwhelming amount of remaining work.

Neither perspective, optimist nor pessimist, is a true representation of the team's status. Usually, the team lies somewhere in the middle. It is the PM's job to navigate these two perspectives, determining the status and building a schedule around it

On multi-year programs, one of the biggest challenges faced by the leadership team is leadership transition. It is rare that a PM or CE can lead for the entire length of these programs. PMs and CEs are often upperclassman or grad students, so graduation must be expected.

Students usually know when they will graduate. Successors should be identified as early as possible, allowing a minimum of one semester overlap. Ideally, the new lead not only supports, but takes over a semester before the outgoing lead graduates. This provides the most realistic training, with the benefit of the past lead as a safety net. Leadership transitions are challenging and can derail the team if not handled properly.

| | The Case of the Disappearing PM | The Case of the Solo PM |
|---|---|---|
| **Situation:** | A team had a senior PM planning to graduate in May. The PM was exploring grad school options, with the possibility of staying at the same university. Because of this possibility, the PI did not seek a new PM or implement a PM transition plan. The PM eventually chose a different university for grad school, and upon graduation, stopped responding to all emails and calls from the team. | A team was experiencing a significant period of leadership transition. Several PMs in a row served short terms. During this, the CE graduated without a trained replacement, followed closely by yet another PM transition. Due to the rapid turnover and minimal training, the new PM was not aware of the need for a CE. |
| **Problem:** | No transition plan was implemented, and everyone assumed the PM would stay. When it was made clear that other options were being explored, nobody was trained as a replacement. Since the PM stopped responding to emails and calls, the team was left without a plan, without continuity, and without leadership. | Without a CE, the RVM was left unmaintained. The team was not aware of many performance requirements and ignored all User's Guide requirements because no one was tracking them. |
| **Solution:** | No student should feel pressured to stay at their university because of a leadership role in UNP. Students with upcoming graduation dates should transition their knowledge to other members, even if they may stay. In addition, the loss of any single person should not be able to break a program. Knowledge should be shared among leadership and team members and thoroughly documented to mitigate the impact of sudden departures. | A subsystem lead was promoted to CE and became responsible for requirements. Unfulfilled requirements were caught which would have been detrimental to ability to deliver. |

Most students will not have prior experience managing a team. This is an excellent opportunity to learn by doing, and most teams become stronger with time as the leadership gains experience. Leaders should seek mentorship from those with more experience and constructive feedback from team members.

The short schedule of the Mission Concept Program means students must stay on task and work aggressively towards deadlines. The PM and CE will encounter difficult decisions such as nixing a mission objective, finalizing a design decision, or debating with the PI. While it is important to make informed decisions, acting definitively allows much faster progress than waiting for an answer to expose itself. This is part of the learning process, and students are encouraged to take ownership of their team. Clear goals should be set for your team as this promotes timely work.

In addition to managing technical efforts, teams are made of people. Leadership will have to manage conflicts and opposing personalities. Leaders should address issues with integrity and avoid micromanagement, and team members should be forthcoming with problems.

Many of these issues are preventable. Regular team meetings help everyone stay aligned, as can other methods of checking in. When onboarding new members, exposure to the big picture and high-level mission requirements help with understanding, and learning their motivation for participating can help leadership assign tasks which are mutually beneficial.

#### 1.5.2.5 Subsystem Teams

Usually, the organization of the engineers who work on the satellite design falls into subsystems like communications, attitude determination and control, and structures. However, what subsystems exist or how those subsystems are organized is up to the university. What one school calls the Communications Subsystem (COM) another calls Telemetry and Command (TNC). One school may have a generalized guidance navigation and control group, while another has a propulsion subsystem, an attitude determination and control subsystem, and an orbital determination subsystem. These delineations are up to the team and are often dictated by PI and student experience and expertise. The key is to ensure all satellite and team responsibilities, components, and capabilities are captured.

#### 1.5.2.6 Subsystem Lead

Most team structures involve one or two subsystem leads per subsystem team. If there are two, responsibilities must be coordinated to avoid gaps. Choosing 1 vs 2 depends on the university, the expertise & strengths of the students, the team size, and the subsystem complexity. The subsystem lead is responsible for the design and test of their subsystem. They must ensure the design meets all requirements, constraints, and needs of the satellite.

The subsystem lead is considered the subsystem expert. While the lead may not make every decision or perform every trade, they should understand everything done in the subsystem. The lead is the person ultimately responsible for the subsystem design; it is critical that they understand it and coordinate subsystem efforts.

It may also be the subsystem lead’s responsibility to divide tasks amongst their team members. Even if the PI, PM, or CE assigns tasks, they will likely need to consult the team lead to understand subsystem status and task availability and priority.

#### 1.5.2.7 Subsystem Team Member

Subsystem team members are usually given discrete tasks or responsibilities. Tasks and responsibilities should be scaled according to the team member’s expertise, capabilities, and time commitment. Because there is such diversity of tasks and responsibilities, team members’ engagement with the team will vary. Regardless of assignment or responsibility, team members must remember to communicate thoroughly and openly to avoid misunderstanding.

#### 1.5.2.8 Common Subsystem Challenges

The biggest challenge faced by subsystem teams is transition of knowledge. Designing, building, and integrating a subsystem requires common understanding from beginning to end. However, with the high rate of turnover among university students, most students will not be around for the entire life of their subsystem on multi-year programs. It is critical that subsystem leads and team members not only transition knowledge to new students, but also properly document the knowledge so it remains within the team.

Another common problem is a lack of communication between team members and leads. Team members need to be open about their availability and commitment to the team and leads need to be clear and direct with expectations. Ambiguity and lack of communication are the source of most subsystem problems.

Perhaps the most difficult problem a subsystem team can face is that of a “disappearing” team member. Attrition of members occurs on most teams throughout the term as student stress grows. Enthusiastic students will often overcommit at the start of a term, then back off as assignments and exams pile up. This is to be expected – and therefore included in planning. A “disappearing” team member is a special case of this, where a team member does not communicate that they are unable to support the team and simply stops attending meetings and responding to emails. There should be a team protocol in place for this occurrence—especially if this is brought on by something less benign than simply quitting the team.

If a subsystem lead feels a situation is out of their depth, they should not be afraid to escalate issues to the PM, CE, or PI. Leadership cannot help if they are not made aware of the problem. The CE should be included in all engineering design decisions, issues, and challenges. Team leadership should foster a culture where members are comfortable discussing issues, mistakes, and challenges openly and honestly.

Finally, it cannot be overstated how important documentation is to the continuity and success of a subsystem. Chapter 4.3. of this document goes into further detail on expected documentation and configuration management processes, but some good high-level rules are:

- Never maintain documents on personal devices. All documents must be in a shared storage so other team members can access it.
- Just because it needs to be documented doesn’t mean it needs to be fancy. Handwritten documents can contain a lot of necessary information. Scanning a handwritten document and placing it in a shared storage counts as documentation.
- Ensure the organization of the document system is itself documented sensible so that future team members can find what they need.

### 1.5.3 Team Participation

It should go without saying that every team in UNP is expected to participate in the objectives of the program. This User’s Guide dictates what we expect, but the following bullets are key points to keep in mind:

- User’s Guide: It is expected that all teams will read and adhere to this User’s Guide and will participate in the program as outlined in this document.
- Reviews: It is expected that the teams will participate in the required reviews outlined in this document as well as any additional reviews that may be required by UNP. Reviews will often result in action items that students must respond to.
- Communication with UNP: It is expected that teams communicate regularly with UNP. This is generally through reviews, presentations, and email correspondence, but may also include bi-weekly meetings or other forms of interaction. The university should provide information, clarifications, trade studies, reports, etc. to UNP if asked.

### 1.5.4 ITAR Policy and Information Control

As US universities and persons, you are subject to International Traffic in Arms Regulations (ITAR). Universities should have an ITAR policy by which students, programs, etc. must abide. In addition, the university will be obligated, per their contract with UNP and the Government, to protect any distribution limited materials such as controlled unclassified information (CUI). While these regulations and obligations exist, UNP attempts to only provide information that is publicly releasable to minimize difficulty for the teams. Materials developed by the university are considered fundamental research and may be released by the university. Materials containing information provided by UNP may only be released if the distribution statement shows they are approved for public release. Note that these distribution marking may not be changed or added to new documents without UNP approval. University documents referencing UNP documents do not need these markings if the sources are approved for public release and properly referenced. Contact UNP and your university contracting and ITAR compliance personnel with questions about information release and university obligations.

Participation of foreign nationals is a common question. As far as UNP is concerned, foreign nationals are welcome, but the PM, CE, and PI must all be US Persons. In these situations, teams must understand and comply with university obligations and federal regulations. When working with non-US person team members, teams should be mindful of the distribution statements listed on materials.

### 1.5.5 Logo and Template Usage

UNP and our partners love that teams are proud to participate in our programs! We encourage you to state that you are in UNP and you're welcome to use our logo, but there are some limitations.

- If there's ever doubt about how a logo may be used, contact UNP.
- The UNP and CSLI logos are controlled by us, meaning the personnel you interact with at these programs can provide permission for their use.
- Agency and Department logos such as NASA, Space Force, and Air Force logos have lengthy documents explaining when and how they can be used. These guides can typically be found online. Discuss with UNP if you feel you must use these logos.
- Logo and template usage must be professional, appropriate, and not misleading. Logos may not be altered in any way.

In general, we encourage you to use our logo for cases such as listing your sponsors, showing that you are part of our programs, or encouraging others to propose to UNP. Our logos and templates should not be used where they make your work appear as if it is created and owned by us. For example, using the UNP slide template for your own slides gives the impression that we are the creators and owners of that material. Avoiding this impression will protect all of us, as information created by the Air Force, Space Force, and NASA is subject to various restrictions. If mishandled or released without appropriate approval processes, it could result in legal repercussions. Best to avoid confusion and make your own school templates.

## 1.6 University Expectations for UNP

A lot has been said about what AFRL and NASA expect of the university programs, but what can you, the university program, expect from us? The responsibilities and services provided by UNP, CSLI, and ER&T are discussed in this section.

### 1.6.1 Funding

AFRL, including AFOSR, NASA CSLI, MDA, and SWAC will provide funding to the university. In the Mission Concept Program, each university is provided a one-time funding drop of a set amount listed in the current RFP. This funding is expected to be used to support the PI and students throughout the program and at required events, such as travel, stipends, and conference fees. Leftover funding may be used at the PI's discretion for equipment, supplies, and other goods and services that support the intent of the program.

### 1.6.2 Internships

In the Mission Concept Program, each university should send multiple students to become interns with UNP. The interns will spend most of the summer in Albuquerque, NM, where various educational exercises will occur in support of the program. The students will also be immersed in a professional engineering environment and will have access to small satellite experts for regular feedback and guidance. At the end of the internship, the students are expected to distribute their knowledge to the rest of their team at their university.

### 1.6.3 Engineering Support

The primary goal of UNP is education, and this is the main area where AFRL supports universities. Multi-day engineering workshops, presentations, recorded lectures, and this User Guide will be provided to the teams. Engineers with extensive spacecraft design experience are available through UNP, as we have access to experts across the Government and industry. Reviews and formal opportunities for feedback also occur regularly. The goal is to help not only the students currently in the program, but to develop strong university teams that exist beyond UNP.

**Note**

The key to leveraging AFRL engineering support is communication. UNP exists to support but can't help if they're not aware of your needs.

## 1.7 Terms, Definitions, and Concepts

Every engineering field has a language. Various terms, definitions, concepts, and approaches are understood throughout the industry. The learning curve is high, but UNP strives to break down barriers to entry. It would be infeasible to explain every concept you'll encounter in the field of small satellites in this guide. This section attempts to explain common terms, definitions, and concepts. Students should ask questions, seek other resources, and know that we're here to help.

### 1.7.1 Mission, Systems, and Subsystems

We will regularly refer to the mission, systems, subsystems, units, and variations of these. It is even possible to have systems of systems. While definitions vary, UNP considers subsystems to be the hardware/software that make up individual, functional sections of a system, while the system is the integrated, more complex combination of these subsystems. Above the system, the mission encompasses everything, even non-technical considerations.

Take a cellphone for example. The "Mission" of a cellphone encompasses a wide variety of considerations but could be summarized to something like "Cellphones provide pocket-sized, reliable, affordable, user friendly communication and information across many methods and domains including talk, text, data, and Wi-Fi." To achieve this mission, a variety of things are needed. In addition to creating a functional cellphone, the mission-level concerns include manufacturability and supply and logistics. The cellphone itself is a system comprised of many subsystems. These include the power supply/battery, the screen, the various antennas and transceivers for Wi-Fi, data, Bluetooth, and near-field communication, the primary computer, the software, and others. Each subsystem must complete its own task and integrate properly with other subsystems. Furthermore, the cellphone system is not the only system necessary to complete the mission. Without cellphone towers and service-provider networks, cellphones would only work on internet. The mission comprises both the cellphone system, and the service system, each with their own subsystems.

In satellites, we generally have the satellite system and the ground system, though complex missions could have additional systems. Each contains various subsystems described below. Every mission is unique, so the subsystems described here should not be taken as a directive, simply as an introduction to common satellite subsystems.

#### 1.7.1.1 Payload

The payload answers the reason for the mission. The data it collects and/or experiments it performs will achieve the mission goals. While this is normally on the satellite, some experiments, such as communications experiments, require a ground system and satellite system to execute the experiment. Sometimes payloads are experimental prototypes of hardware for other subsystems, such as radios or thrusters, while many times they are sensors such as imagers,

#### 1.7.1.2 Electric Power Subsystem (EPS)

The EPS performs a few critical functions. It normally includes power generation, such as solar panels, energy storage, such as a battery, and power regulation, distribution, and switching to the rest of the satellite.

#### 1.7.1.3 Command and Data Handling Subsystem (CDH)

The CDH subsystem includes the primary satellite on-board computer (OBC). Some teams include software development within the CDH subsystem as well.

#### 1.7.1.4 Software Subsystem

Software may encompass ground and flight software, may be placed under CDH, or may be organized in another manner. Regardless, software requires constant, focused effort and must remain a priority.

#### 1.7.1.5 Attitude Determination and Control Subsystem (ADCS)

The ADCS is responsible for determining the attitude (orientation) of the satellite, as well as controlling the attitude. This is sometimes known as Guidance, Navigation, and Control (GNC) which is a broader term. GNC would encompass ADCS, propulsion (if applicable), and a global navigation satellite system (GNSS) receiver such as the Global Positioning System (GPS). The ADCS typically includes sensors, to collect attitude data, a computer board and software algorithms to compute an attitude and control outputs, and actuators to control the satellite attitude. Common ADCS sensors include magnetometers, to read the Earth's magnetic field; coarse and fine sun sensors, to detect the direction of the sun; and star trackers, to detect known stars. Actuators typically include magnetorquers, which push against the Earth's magnetic field, and reaction wheels, which accelerate a spinning mass on a motor to provide torque.

#### 1.7.1.6 Propulsion Subsystem (Prop)

UNP discourages propulsion systems on university satellites. Most CubeSat missions have no need for propulsion, and it drastically increases complexity and safety concerns. There are also very few thrusters available on the market that fit CubeSat form factors. A propulsion unit would generally fall within the GNC subsystem.

#### 1.7.1.7 Communications Subsystem (COM)

The COM subsystem includes the radio(s) and antenna(s) used for command, control, and data downlink to/from the satellite. Note that a GPS antenna, an experimental payload radio/antenna, or other communications-related hardware not used for command, control, and data downlink would generally fall under other subsystems such as ADCS for the GPS antenna and Payload for the experimental system.

#### 1.7.1.8 Structures (STR)

The structure, typically made of machined aluminum, mechanically supports all other subsystems, ensuring the spacecraft can survive launch, orbit, and other stresses on the vehicle.

#### 1.7.1.9 Thermal (THM)

The Thermal subsystem is typically quite simple on small satellites. It typically includes temperature sensors, heater(s), and may involve metallic heat straps or heat sinks, as well as selected exterior coatings on the satellite structure. Large satellites may have sophisticated, active heating and cooling systems, but small satellites tend to be entirely passive except for a battery heater.

#### 1.7.1.10 Ground System/Subsystem (GND)

The ground subsystem includes multiple disparate parts, which may be organized as separate subsystems. The ground station(s) includes the radio(s), antenna(s), and other on-site hardware such as amplifiers and modems. The ground data system includes operations centers, data processing and exploitation systems, and networking with distant ground stations if using ground station service providers.

#### 1.7.1.11 Ground Support Equipment (GSE)

Ground support equipment includes mechanical and electrical equipment used for direct interaction, support, and test before the satellite launches. Mechanical ground support equipment (MGSE) may include shipping containers, integration stands, and other useful parts that are not part of the satellite. Electrical ground support equipment generally includes power supplies, a computer, and various switching and data collection systems. It provides electrical access to the satellite once fully integrated for tasks such as charging batteries, updating software, and verifying functionality.

### 1.7.2 Satellites

Satellites come in all shapes and sizes and perform a wide variety of missions. Small satellites are usually defined by a mass of less than 500 kg, while mini, micro, nano, pico, and other satellite prefixes divide the general small satellite term into narrower categories. CubeSats are within the class of small satellites, typically fitting into the nanosatellite mass range. In addition to mass limitations, CubeSats follow standard dimensions found on many launch vehicle deployers, making it much easier to find a launch. A standard unit or “U” is approximately 10x10x10 cm, though standards vary slightly. These units can be stacked together for larger spacecraft, with common sizes being 1U, 2U, 3U, 6U, and 12U. UNP and CSLI, accept satellites of these sizes due to the ease of launch compared to non-standard satellites.

### 1.7.3 Orbits

The most common Earth orbit terms are low Earth orbit (LEO), medium Earth orbit (MEO), and geostationary, or geosynchronous equatorial orbit (GEO). These orbital regimes cover the space around Earth out to GEO which is around one tenth of the distance to the moon. LEO encompasses everything below 2000 km in altitude and is where most spacecraft reside. Nearly all CubeSats are launched to LEO in the 350 – 550 km altitude range. In this orbit regime, spacecraft move around 7.5 km/s relative to an observer on the ground, circling the Earth with an orbital period of roughly 90 minutes. This greatly influences the ability to talk to a spacecraft via radio, as even a perfect, direct overhead pass of a ground station only provides around 10 minutes where the satellite remains within line of sight. Another significant consideration in low orbits is atmospheric drag. Satellites will be slowed over the course of months or years and experience slight torques due to the extremely thin atmosphere in this orbital regime. Finally, satellites in these low orbits are within the majority of the

Earth's magnetic field. This provides some protection from charged particles and solar wind and can also be used for attitude control via electromagnets known as magnetorquers.

MEO stretches from 2000 km up to 35,786 km altitude. This is a less common orbit range partially due to the presence of Van Allen radiation belts. A notable exception is that nearly all global navigation satellite systems, such as GPS, exist in this regime at an altitude of about 20,000 km. A benefit of this altitude is the high stability and predictability of the orbit due to no atmospheric drag and minimal gravitational field perturbations. GPS satellites orbit the Earth every 12 hours and require extremely accurate position knowledge to pass to GPS receivers on the ground.

Finally, Geosynchronous orbits exist at 35,786 km. This is a special altitude where satellites orbit Earth in exactly 24 hours. Satellites in inclined geosynchronous orbits will trace a figure 8 over the surface of the Earth, while those in the GEO belt, the ring above the equator, stay fixed over a point on Earth. This is highly beneficial for communications, as satellite dishes can point to the sky with no need for slewing. For reference, the Earth has a diameter of around 12,765 km, meaning the GEO belt is around 3x the diameter of the Earth away from the Earth.

Orbits may be defined in a variety of ways, but two common methods are Keplerian orbital elements and two-line elements (TLEs). Keplerian elements provide easy to understand parameters for defining an orbit, while TLEs are much less digestible by humans, but much more so by orbit modeling software. There are six Keplerian elements. Three of these, the apoapsis, periapsis, and inclination, provide most of the information needed to model trends such as solar power, earth coverage, and satellite temperature in common CubeSat orbits. The remaining three, not described here, are the longitude of the ascending node, argument of periapsis, and true anomaly. Due to physics, orbits take the shape of an ellipse, where the body being orbited exists at one of the foci. The high point of the orbit ellipse is the apoapsis, and the low point is the periapsis. An alternative to specifying apoapsis and periapsis is to specify the length of the semi-major axis and the eccentricity. A set of two of these parameters provides the first two Keplerian orbital elements.

**Note**

Periapsis and apoapsis are generic terms for the highest and lowest points of the orbit. You may have heard the terms perigee and apogee used in this manner. These are the same, however, the "gee" suffix applies only to trajectories around Earth. Orbits around the sun have a perihelion and aphelion and those around Jupiter have a perijove and apojove. Periapsis and apoapsis may be used for any central body.

The elliptical shape provided by the first two elements makes an orbital plane. The orbital inclination is the angle between Earth's equatorial plane and the orbital plane of interest. Note that between 0 and 90 degrees, the inclination angle is the same as the maximum Earth latitude over which the satellite will pass.

The remaining elements allow an orbit to be completely specified. While not addressed here, students should familiarize themselves with common orbit terms and basic mechanics, for which extensive resources can be found online.

## 1.8 Conclusion

The rest of this guide explains UNP in detail. It is expected that the team will be familiar with this guide throughout the program. As you progress, remember that the main purpose of UNP is to educate. While you work hard, remember to have fun, and learn along the way!

## 2 Space System Development Process

This chapter provides a brief overview of common space system development processes. In many cases these are widely accepted practices across government and industry, in other cases UNP provides our approach which has been adapted to the educational environment. This chapter will also provide context for many UNP requirements. Note that the UNP Nanosatellite User Guide has a similar, but more detailed version of this chapter.

Note that while this chapter generally refers to the satellite, this process must be applied to the entire space system. A space mission and system not only include the satellite, but also the ground station(s), ground data system, and other systems or subsystems that contribute to the mission as a whole. These must not be ignored as they are critical to the completion of the mission. Teams that wait to design ground systems or other critical non-spacecraft pieces until late in development regularly find themselves with difficult, expensive, and time-consuming rework.

### 2.1 Mission Inception

Satellite design begins with an idea. A scientist or engineer, usually the mission's Principal Investigator (PI), discovers a problem or situation that only a mission to space can solve. This idea cannot be solvable on the ground due to the expense and difficulty involved with space. The PI must understand and be able to defend the need for space.

Usually, this idea evolves into the satellite's mission statement. The mission statement is a one or two sentence statement of the entire mission's purpose, usually focusing on the science or technological objectives of the mission.

While employer-dependent, there is typically a process the PI must follow to propose the idea to other members of the organization and gain support. If the mission makes it through this process, the PI can assemble a team to begin satellite development.

| How does this relate to UNP? |
|---|
| In UNP, this phase of the satellite design occurs in the proposal process and during the months leading to the system concept review (SCR). If you're participating in UNP, your PI responded to our request for proposals and your mission was selected. While selection suggests the mission is meaningful and necessary, it may also mean the selection committee saw potential in the mission, but refinement is needed.<br>At the system concept review, UNP helps each school refine their concept. |

### 2.2 Systems Engineering

There is a common misconception that once the mission statement is defined, the program can start selecting hardware and building their satellite. This is not the case. UNP follows a systems engineering process which requires additional thought and analysis effort up front but provides greater chance of success.

Systems engineering is engineering with the whole system in mind. For example, a satellite is made of several subsystems. You might have a command and data handling subsystem, an electrical power system, an attitude determination and control subsystem, and so on. In isolation, the engineers in charge of these subsystems would design the most robust subsystem possible. An ADCS engineer may choose high power control moment gyroscopes and multiple star trackers. These components would undoubtedly meet the needs of the subsystem, however, concerns such as their large size or high-power draw were ignored. A satellite designed for a science mission would have no room or power for additional components. This is not systems engineering.

Systems engineering takes a holistic approach. Each subsystem is part of a greater system and must be designed to complement other subsystems and the satellite. Good systems engineering also recognizes that the satellite is just one system within the greater mission. The ground station and satellite are both parts of the mission. Neither should be designed in isolation or the mission is likely to fail.

Systems engineering also provides a framework for engineering management across the entire mission lifecycle, including concept definition, detailed design, and verification. If the ambitious ADCS engineer from earlier had completed a trade study from a systems engineering viewpoint, power and volume would have been considered, resulting in the ADCS most appropriate for the whole system.

There are many types of systems engineering, each tuned for a specific organization or effort. UNP loosely follows the approach taken by the NASA Systems Engineering Handbook, but alters it based on differences in the application and lessons learned. Universities are very different organizations than NASA. Funding levels, mission complexity, risk posture, team size, and countless other aspects are incomparable. UNP is also an educational program, so beginning with a simplified approach to systems engineering promotes learning. In the end, the UNP approach to systems engineering is not the only valid approach, and in many cases would not be the correct approach. It provides students with a strong introduction to systems engineering which will be applicable, though not identical, to nearly all industry methods,

All systems engineering approaches share common goals. The intent is to take an idea from a stakeholder, clarify the pieces of the idea that hold meaning and value, ensure the mission is feasible, then design, build, test, and operate a mission in such a way that it can demonstrably achieve the initial goals of the mission. Various approaches, such as the “V” model of systems engineering, NASA’s Systems Engineering Engine, and others provide frameworks for this process. You may also encounter concepts such as model-based systems engineering (MBSE). As engineers, we want to immediately begin CAD design, board layout, and code development, but if we don’t know the required performance of the systems we’re designing, how can we be sure the design is correct? Systems engineering exists to avoid wasted effort and expense of designing a system without proper understanding.

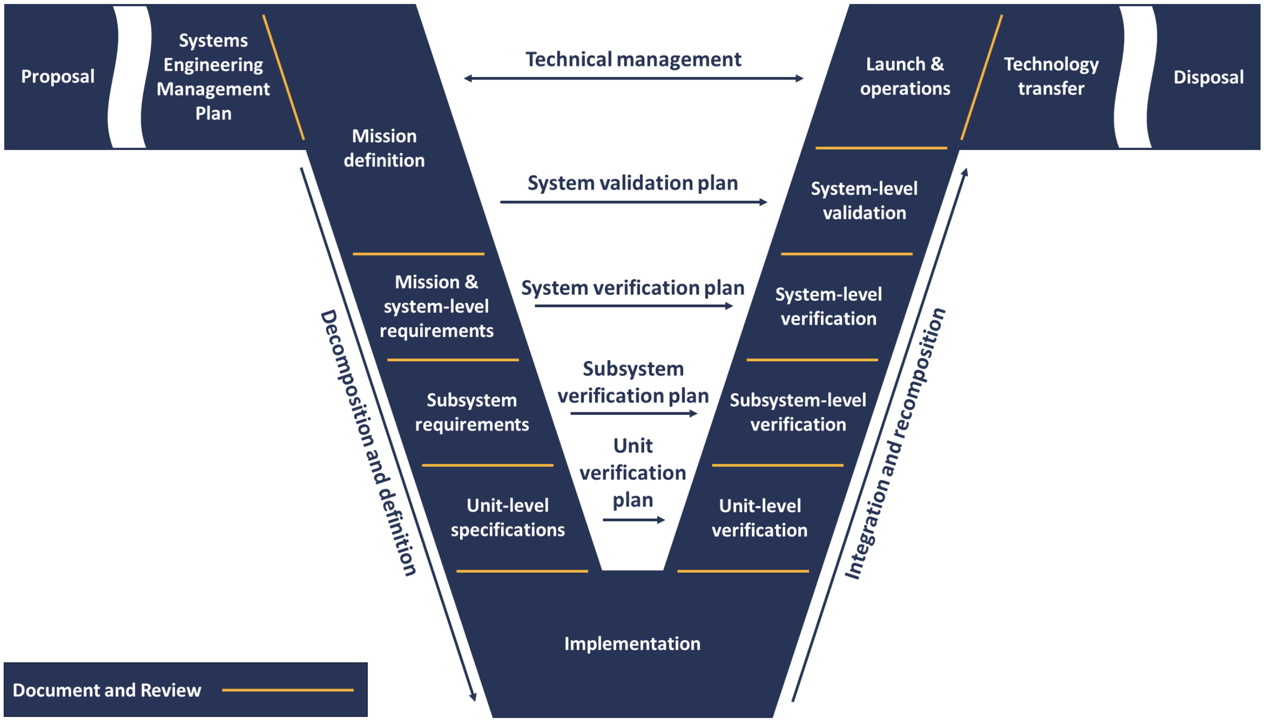


Figure 2-1. The "V" Model of Systems Engineering

## 2.3 Mission Concepting

All missions start with an idea, but usually the idea is not ready for a mission. Extensive critical thinking must be applied to the idea, ensuring it is meaningful and worthwhile. Those with interest and funding, known as stakeholders, must clarify exactly what they want from the system with quantitative figures of merit. Achieving these definitive statements of purpose can be a challenge, and requires feasibility and performance analyses, trade studies, and lengthy discussion between the stakeholders and engineering team. While not an ideal flow down, it is common to find systems engineers determining quantitative metrics for indecisive stakeholders to debate and eventually agree upon. This initial effort can be highly frustrating. It involves managing not only technical decisions, but people. Stakeholders may have conflicting or lofty goals and engineers will push back with difficulty and cost insight. This process generally feels quite chaotic, as though nobody agrees. However, at the end of this effort the engineering team is left with a well-defined, quantitative set of design goals. If the team achieves those goals, then everyone involved should be happy with the mission results. Ignoring this process can lead to misunderstandings and lack of stakeholder satisfaction much later in the mission lifecycle when issues are much harder to resolve. These figures of merit can also be helpful late in development if making hard decisions about cutting performance. Poorly defined metrics will not allow clear determination of overall mission/system impact due to subsystem performance changes. A recommended method to guide this process is to outline the “Final Report” and desired information output for the mission. In UNP, we help teams develop a mission statement, mission objectives, and mission success criteria, which are the formal statements of the mission goals.

**Note**

UNP follows the Mission Statement > Mission Objectives > Mission Success Criteria approach. While most systems engineering processes define mission statements and objectives, success criteria may go by another name or form (i.e., a target value with acceptable thresholds around it). The key is to develop well-defined, quantitative, mission-level requirements allowing for system design.

A common term you'll hear in systems engineering is flow down. The idea is to take the mission statement, which is very broad, and create from it defined statements of qualitative mission goals, known as the mission objectives. Then the mission objectives are flowed down into mission success criteria which are quantitative statements of things the satellite must accomplish. These high-level goals then flow down into system level requirements, or things the system must be designed to do at a high level, and so on all the way down to component requirements – possibly even lower depending on the mission.

Good flow down makes it easy to see how changes to one system affect other systems. It also helps ensure requirements are not missed, conflicting, duplicated, or excessive. Finally, it provides a framework for ensuring the system achieves its goals. Testing can begin at the lowest level of the flow down and progress upwards, ensuring more complicated tests achieve higher level requirements after the building blocks are functional. Common levels in a university mission flow down include Mission, System, and Subsystem. Sometimes, a subsystem may have unit or component-level requirements beneath it, and rarely these may even flow down to piece part requirements. Not every mission, system or subsystem must have the same depth of flow down, it should be tuned to the complexity and needs of the use case. Requirements are about communicating design needs (flow down), and design capability (verification, flow up). This concept is demonstrated in Figure 2-2.

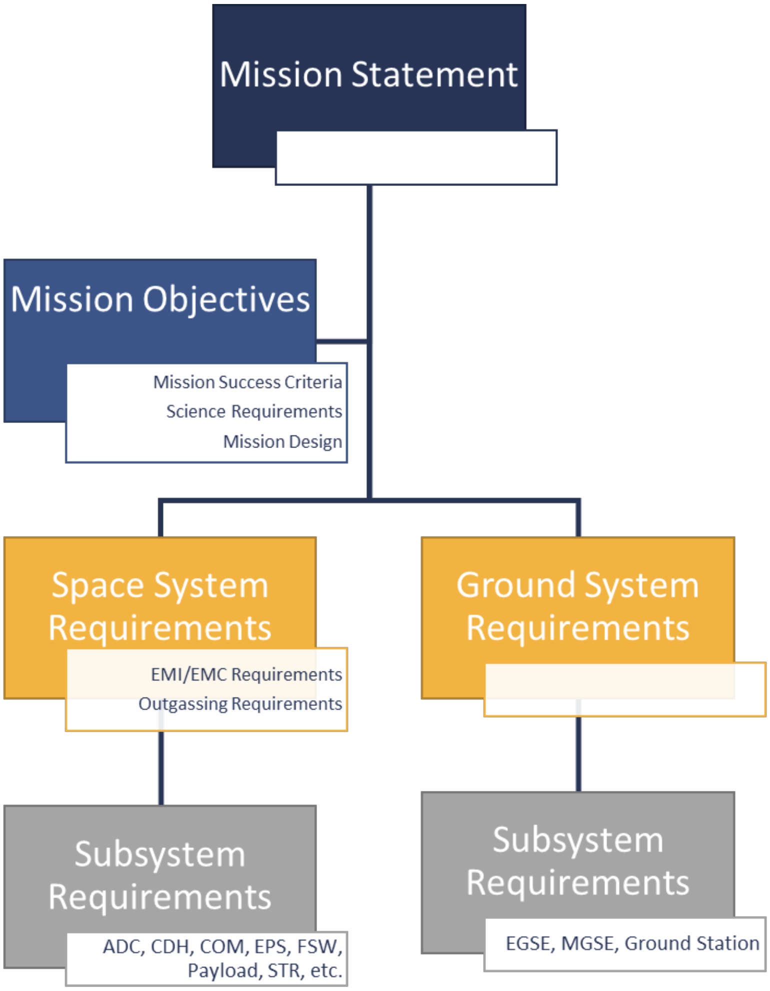


Figure 2-2 Mission Flow Down

### 2.3.1 Constraints

While an ideal systems engineering approach might be completely requirements based, this is never a reality. All missions will encounter constraints – limitations imposed on the mission. This is especially true in the university environment where budgets, personnel experience, and facility capabilities may be limited. Constraints should be identified early to ensure the design is achievable. These will not necessarily flow down like requirements but must be documented and tracked at the appropriate level. For example, your university may already have a working Ultra High Frequency (UHF) ground station and may not have the funds to build an S-Band ground station or purchase ground station services. Therefore, you are constrained to the use of UHF on your satellite and ground station. While sometimes a constraint may fit in the flow down of your system, maybe UHF was all you ever needed, other times it may not. These conflicts must be thoroughly understood and justified to ensure the fully realized system achieves its goals. Clearly include the constraint, source, justification, notes, and other relevant details at the appropriate level.

### 2.3.2 Writing Requirements

Requirements are easy-to-understand statements that express clear needs, not solutions, of the satellite. They are not arbitrarily written but are formed through thorough understanding and analysis of the mission science and goals. Writing requirements is not easy and is often best approached as a team effort. All requirements development should follow a few guidelines which are described in this section.

Requirements must be verifiable, achievable, consistent, necessary, and unambiguous. Verifiable means we must be able to test, inspect, demonstrate, and/or analyze the system to prove that it meets the requirement, meaning requirements must be quantitative statements, not qualitative. Words like *sufficient*, *enough*, *sometimes*, and *often* are red flags that a statement may be qualitative, not quantitative. Achievability is also a concern. While this may seem obvious, it is not uncommon to find inexperienced assumptions or stakeholder desires leading to unachievable requirements. Requirements must be based on understanding and analysis, not values that seem reasonable at the time. Consistency refers to requirements not conflicting across systems, subsystems, or levels, and to ensuring lower tier requirements properly flow from higher tier requirements. Requirements must be necessary, not only to avoid unnecessary work, but also to avoid confusion and conflicts. Finally, requirements must be unambiguous, ensuring they can only be interpreted one way.

Requirements must be thoroughly documented. A common way of doing this is a requirements verification matrix (RVM). Typically captured in a spreadsheet, an RVM can be organized into tiers, broken down by system and subsystem, and accompanied by identifiers, notes, status, and other necessary tracking data. An example requirement is shown in Table 2-1.

**Table 2-1 Example Requirement**

| Number | Requirement | Source | Verification Method | Status | Verification Document | Justification | Notes |
|---|---|---|---|---|---|---|---|
| ADCS-1 | The satellite shall be capable of three-axis stabilization. | MO-2 | Analysis | 🟥 | ADCS-D-001 | To track transient objects, the satellite must perform controlled slews in all three axes | See document “ADCS-C-003 Pointing Budget” for analysis |

### 2.3.2.1 Identifier

Every requirement should have a unique identifier. It should provide shorthand for finding and organizing each requirement. In the example above, the number is ADCS-1. This notation is because the requirement falls in the Attitude Determination and Control Subsystem (ADCS), and it is the first requirement listed in that section. Identifiers will vary based on the requirement's tier, sub-tier, and subsystem. For example, the second Mission Success Criteria might be called MSC-2. Identifier systems may be unique to each school if they serve the necessary purpose.

### 2.3.2.2 Requirement

In addition to the above guidance, requirements should always be written as "shall" statements, as opposed to "should" or "will." Shall means that a requirement must be done and must be verified. It is ok, early on, to create a requirement with a value that is to be determined (TBD), to be refined (TBR), or to be confirmed (TBC). These are used as placeholders for a future, better vetted, value. Not every analysis will have an answer right away, these placeholders allow for clear flow down while the RVM is still in development. By SRR these should be used sparingly, and ideally should be replaced with final values by the preliminary design review (PDR).

### 2.3.2.3 Source

The requirement source tells us the higher-level requirement or goal from where this requirement flows. In the example above, Mission Objective 2 (MO-2) may state something like "The spacecraft shall be able to track moving objects in low Earth orbit (LEO)." Since it must track moving objects, it therefore needs three-axis control. It wouldn't be able to meet this mission objective while tumbling or even with only two-axis control. Therefore, ADCS-1 was derived from MO-2.

Not all requirements will cleanly flow down. Some requirements will come from external sources such as UNP, a Launch Vehicle (LV) Interface Control Document (ICD), or a CubeSat dispenser user guide. Requirements such as this, as well as constraints, may not cleanly fit the flow down, but they must be integrated into requirements tracking in a manner that ensures conflicts are avoided. They can be entered at the appropriate tier and should have clear justification and notes sections to provide additional context.

### 2.3.2.4 Verification Method

Every requirement put in the RVM — whether a User's Guide requirement, dispenser requirement, or a university constraint — must be verified. That is, the team must determine that their satellite, ground station, and other systems in the RVM are compliant with the requirement. There are four standard methods of verification: inspection, demonstration, analysis, and test.

- Inspection is when a value can be directly measured or shown (e.g., mass, dimensions).
- Demonstration is when a functionality or capability can be shown as working, typically in a situation where the only possible states are "Works" or "Doesn't work."
- Analysis involves mathematical, simulation, or other form of assessment to show that a system is capable of the proper function (e.g., link budget). This often works in conjunction with test data for thorough verification.
- Test involves exercising the hardware/software of interest to show it performs as intended. This is the most common method. (e.g., sending software commands to the satellite and ensuring the responses and behaviors are correct).

### 2.3.2.5 Justification

Good requirements are based on analysis and reasoning. When created, they make sense to the team, but in the future, student may have no idea why a requirement exists or how a specific number was determined. Providing a justification for each requirement ensures stakeholders, future members, and reviewers can understand the origin of a requirement. Linking/referencing any analysis upon which a requirement is based is good practice and encouraged.

#### 2.3.2.6 Notes

There may be additional, relevant information not captured in one of the above sections. If so, include it here. A little extra documenting effort up front will save a lot of time and frustration as future students try to understand prior choices.

### 2.3.3 Tier 1 Requirements

The mission objectives and success criteria are the first requirements developed. The mission statement should be broken into clear descriptions of separate mission goals, called mission objectives. While these are much more definitive than the mission statement, they may still be qualitative. To thoroughly define the mission goals, generally in terms of actions the spacecraft must complete, we go one step further.

Mission success criteria are quantitative statements of what the mission must accomplish (i.e., what do you need to learn, understand, see occur?). Generally broken into minimum and full success criteria, these define mission success or failure. Take the mission objective used as an example above, “The spacecraft shall be able to track moving objects in low Earth orbit (LEO).” This is a qualitative statement with numerous unknowns. LEO objects move at thousands of miles per hour in many different directions, and could be incredibly close, or thousands of miles apart. If two objects with high relative velocity are close together, the slew rates necessary for one spacecraft to track the other could be impossible to achieve. Similarly, if the system is using optical tracking the object must be bright enough to be visible. There must be quantitative bounds on the goal of tracking LEO objects, as well as the number of times this must be performed to call the mission successful. A minimum mission success criteria flowing from this objective could therefore be, “The spacecraft shall track a minimum of 1 LEO object of magnitude +10 or brighter for at least 30 seconds with the observer slew rate no greater than 2 degrees per second per axis.” The full success criteria may be similar but might increase from tracking one object to tracking 10 across the spacecraft lifetime. If the mission achieves the minimum success criteria, we can rest easy in the knowledge that we did not fail our mission.

The mission objectives and success criteria are the top tier(s) of the RVM, typically tier 1. There is no defined way to create an RVM, just ensure it is organized in a sensible manner. These mission-level requirements, if changed, would affect the entire mission. Several types of requirements can go into this top section. Mission statements can be multiple, compound sentences, while mission objectives often break this into a few discrete, simple segments. Mission success criteria are also discrete statements, providing quantitative metrics for the mission. A good rule of thumb is that no objective, criteria, or requirement should contain more than one verb.

### 2.3.4 Tier 2 Requirements

Tier 2 requirements are usually system level requirements. Every mission will contain multiple systems. Even a single-spacecraft mission will have a satellite system and a ground system. A proximity operations mission may have two unique satellite systems and a ground system. Each system requires its own system level requirements.

**Note**

How do science requirements flow through the RVM?

Say your mission statement relates to understanding atmospheric density. To fulfill this requirement, your satellite may collect data from a neutral mass spectrometer. To fully characterize the atmosphere, you calculate it will take 100 orbits between 400 and 500 km altitude.

The required science measurements have now determined your orbit lifetime and required altitude range. The science collections don't only inform orbit requirements. Maybe you can take readings at any attitude, so attitude control is unnecessary, but it's very important that you know exactly how readings were taken, which means you need very precise attitude determination. Or maybe the volume of data from the payload drives you to specific satellite computer requirements.

The purpose of the satellite is to fulfill the science mission, so nearly all subsystem requirements will be impacted by it.

UNP provides numerous requirements, mostly system-level, to UNP teams. One example is the outgassing requirement. The term "outgassing" describes a phenomenon where certain materials, when introduced to the vacuum of space, begin to evaporate, or sublimate. The concern associated with this matter leaving the satellite is not only structural, but also that the matter can redeposit, on critical areas of the satellite, like lenses. Every component going to space must have limited outgassing properties, making it a system level requirement.

In addition to the UNP-levied requirements, most programs will have mission-specific system level requirements belonging in Tier 2.

### 2.3.5 Tier 3 Requirements

Tier 3 requirements are generally subsystem requirements. Each subsystem will have its own set of requirements at this level. Depending on the mission, subsystem, and team structure, it is possible to have further tiers below subsystem-level.

**How does this relate to UNP?**

In UNP, RVM development begins after your mission science is understood. The initial revision is submitted for the system requirements review (SRR). The requirements in your RVM will drive your design, making the RVM critical. At the SRR, UNP provides feedback on the requirements, typically resulting in a complete RVM overhaul. It is critical that the students fully define their requirements before beginning design.

### 2.3.6 Analysis and Iteration

Now that many systems engineering and mission concepting topics have been explained, how does one develop a mission? While there are linear development models, such as the "V" model of systems engineering, these ignore the realities of iteration. The V is a useful guide for the large steps, but don't assume you can ignore prior steps once you start subsequent ones. Throughout the lifecycle of a spacecraft there will be times when revisiting earlier steps is necessary. During mission concepting, this iteration is rapid and expected.

As mentioned earlier, UNP loosely follows the approach taken by the NASA Systems Engineering Handbook. The approach to mission concepting is shown in Figure 2-3.

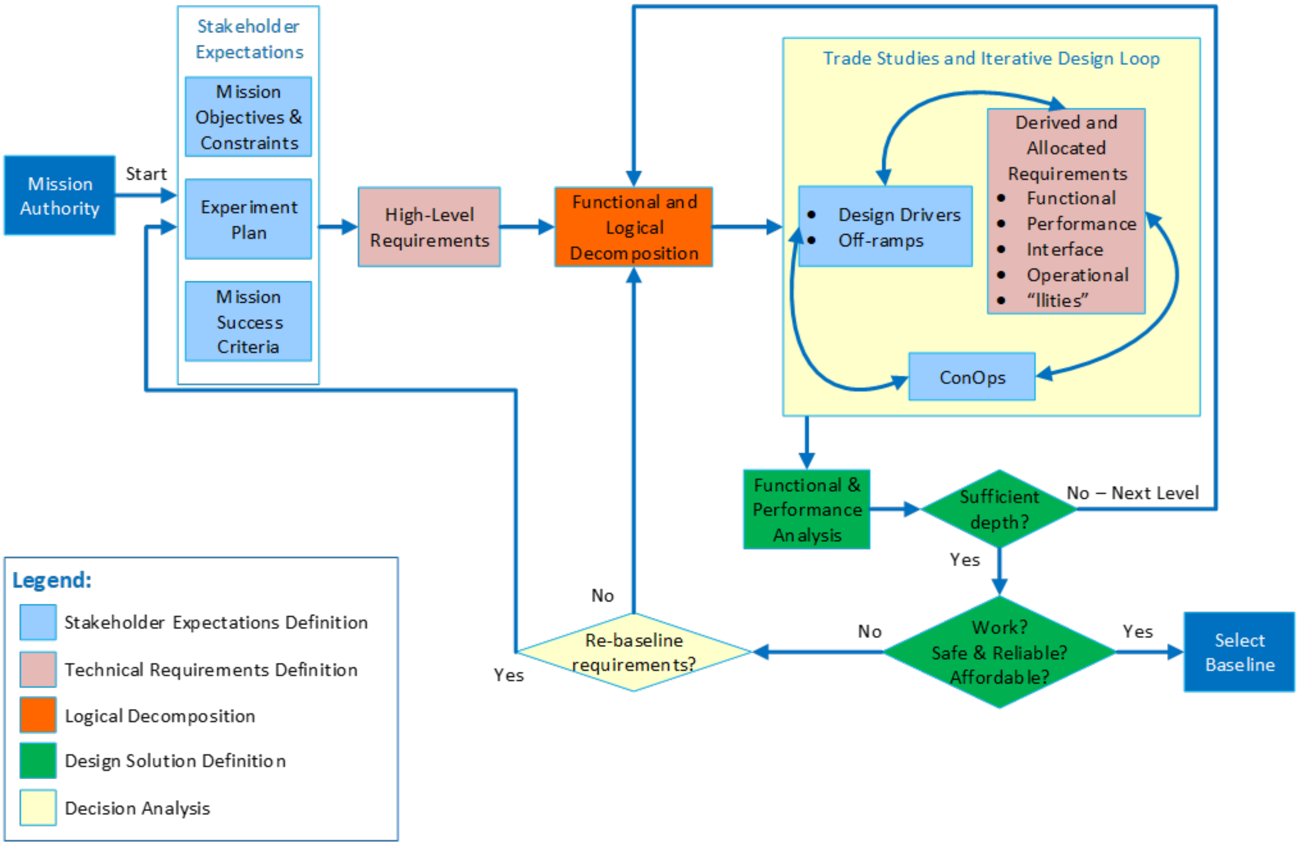


Figure 2-3. System Design Process Relationships from NASA Systems Engineering Handbook

This diagram much more accurately represents the iterative nature of mission concepting. The ability to return to an earlier point and improve the system based on new information exists not only within each step, but also across steps. The team will determine initial goals, but the feasibility of those goals will require analysis and understanding of the system. For example, a radar mission will require a lot of power, but the desired CubeSat form factor may not be able to generate enough. Based on the payload power draw, what size solar arrays will be needed for the CubeSat? In addition, the data generated by the radar will need to be downlinked. Radio transmitters with high data rates draw a lot of power, so the power generation subsystem must be sized for the entire satellite, not just the payload. Basic analyses such as these will be performed, which may result in updated mission goals. Perhaps the updated mission goals now affect separate subsystem analyses. To continue the prior example, perhaps, the necessary solar panels fit on a CubeSat, but are very large and therefore require a complex folding and deployment mechanism. Complicated mechanisms may not function properly, so we've introduced a risk that our panels never properly deploy, limiting the available power on orbit. In addition, large deployable systems will affect the inertia and rigidity of the satellite, creating concerns for the attitude determination and control system which must be addressed. We should certainly revisit every analysis impacted by changes, but we could also approach this problem through the concept of operations (CONOPS). Instead of requiring excessively large solar arrays, maybe we reduce average power usage by operating the radar less often. The satellite could store energy in a large battery which discharges as the radar consumes power beyond that provided by the solar arrays.

During this process, basic analyses should be performed for all relevant systems and subsystems. Highly detailed analysis and design will occur later in the design process, but these initial versions should be accurate enough to show feasibility and broad system characteristics. Developing a baseline set of mission goals and requirements typically takes 4+ months for UNP missions and may take years for large Government missions. Some of the common factors and subsystems which must be addressed during this time are described below. Note that this is not an exhaustive list, and every mission is different. Teams must identify the analyses required for their own mission. In addition, this list is not in order of importance or priority, those also depend on the mission.

#### 2.3.6.1 Payload

The data the payload collects will achieve the mission goals. The team may be constrained to a specific payload from the start or may have the ability to push back on various design aspects. Regardless, understanding the payload requirements (power, pointing, data, etc.) will be critical to performing the rest of the necessary analyses.

#### 2.3.6.2 Electric Power Subsystem (EPS)

The power subsystem normally includes power generation, storage, regulation, and distribution. Some critical pieces to consider include power generation and battery size. These should be checked for a variety of expected orbits and attitude states, including tumbling. The critical tools of the power engineer are the power and energy budgets. The power budget ensures the generation, storage, regulation, and distribution systems can accommodate necessary power draws. The energy budget looks at the energy in and out of the battery over time as the spacecraft passes in and out of the sun, adjusts pointing, turns components on and off, or performs any other operations.

#### 2.3.6.3 Command and Data Handling and Software Subsystems

The critical pieces which must be addressed in early analysis include software architecture, data budgets, and computing budgets. Ensure the expected architecture is achievable in software. This may be as simple as talking with a software expert, but software should never be ignored as a task for later in development. The data budget collects information on data generation rates and amounts across the satellite and ensures it can not only be stored, but also downlinked based on radio data rate, ground station access times, and orbit choices. A computing budget ensures processing power is adequate for data processed onboard.

#### 2.3.6.4 Attitude Determination and Control Subsystem (ADCS)

The ADCS is responsible for determining the attitude (orientation) of the satellite, as well as controlling the attitude. While determination, or knowledge, is necessary for control, some satellites only need good knowledge and not much control, while some may not even have an ADCS. The pointing and momentum budgets are the primary tools used in this subsystem. The pointing budget collects error sources and ADCS performance specifications and compares them to the quasi-static pointing needs of the mission or other subsystems. The momentum budget collects inertias, actuator performance and momentum specifications, and slew rate requirements for a similar comparison in dynamics.

#### 2.3.6.5 Propulsion Subsystem (Prop)

Teams are highly encouraged to avoid propulsion systems on their satellites. If propulsion is necessary, considerations include safety of propellants and pressure systems and increased operations security. A delta-v budget is used to evaluate maneuver capabilities.

#### 2.3.6.6 Communications Subsystem (COM)

The communications subsystem must ensure the chosen radio, antenna, ground station, orbit, and pointing characteristics of the satellite allow adequate communication between the ground and the satellite. This is done with a link budget, balancing transmit power and gains with losses, receiver sensitivity, modulation, data rates, noise, and other adverse effects.

### 2.3.6.7 Structures and Thermal (STR and THM)

While eventually the structures subsystem will be responsible for a highly detailed design, this is not necessary during early analyses. The goal in this case is to determine a form factor and basic layout. The first step is likely a volume model, where rough models of the components are arranged in a sensible manner to ensure fit. The mass budget, ensuring the total mass of the satellite does not exceed allowed deployer mass, will also be important, but is typically not a limiting factor for CubeSats. Most educational satellites are CubeSats which follow standard dimensions and masses to fit launch vehicle deployers, and therefore make it much easier to find a launch. A standard unit or “U” is approximately 10x10x10 cm, though standards vary slightly. These units can be stacked together for larger spacecraft, with common sizes being 1U, 2U, 3U, 6U, and 12U. UNP and CSLI, accept satellites of these sizes due to the ease of launch compared to non-standard satellites. The volume model should determine the necessary form factor. When developing a model, care must also be taken to ensure component connectors and apertures are oriented and sized sensibly. For example, a spacecraft with some solar panels, an antenna that must point at the ground to communicate, and a payload that looks at the earth horizon should orient these components on faces which normally point toward the desired target to minimize slewing. The structures personnel may also be responsible for thermal modeling and analysis. At this stage, a thermal model does not need to be completed, but components that are temperature sensitive or output a large amount of heat should be noted and managed appropriately.

### 2.3.6.8 Ground System/Subsystem (GND)

The ground subsystem includes multiple disparate parts, which may be organized as separate subsystems. The ground station(s) is closely tied to the communication subsystem and must work closely with them on link budgets, access times, and ground station choices. The ground data system is the other piece, which includes operations centers, operations/command and control software, data processing, storage and exploitation systems, and networking with distant ground stations if using ground station service providers.

### 2.3.6.9 Ground Support Equipment (GSE)

Ground support equipment includes both mechanical and electrical elements. For CubeSats, the mechanical equipment is often structurally simple, but still requires significant thought. It must be easy to handle and avoid damaging the spacecraft during lifting and set up activities. Electrical support equipment must not only be designed for safe operation with the satellite hardware but may also support development and prototype efforts as well as the flight system. Anything that interfaces with flight hardware should be treated like flight hardware to mitigate risk of harm.

### 2.3.6.10 Orbit Design

Orbit design involves choosing an appropriate orbit or range of orbits for the mission. There may be an ideal orbit, providing the best possible data, but most likely this is not the only orbit that will work. It is important that the acceptable range is reported in the RVM, though the ideal could be added to the notes. Small satellites are usually secondary payloads, they do not get to dictate their orbit but must accept rides on other launch missions. Knowing what is and is not acceptable must be tracked, not just ideal.

**Note**

Small satellites are secondary payloads, they don’t drive the choice of orbit. A wide range of acceptable orbits, including altitude and inclination, is critical to securing a launch. A satellite may desire a 90˚ inclination for maximum data, but if they get data at as low as 60˚ then the range of 60˚- 90˚ would be the acceptable orbital range. Understanding how the science objectives relate to orbit selection is critical to the design of the satellite.

Other items to be considered in orbit design include system and policy constraints and ground station location. For example, the amount of time it takes to passively de-orbit is directly related to the orbital altitude. Orbital debris laws exist which limit the time a satellite may remain in orbit. This requirement will either drive the upper bound of orbital altitude or dictate the need for a de-orbit mechanism. Ground stations must be within line of sight of the spacecraft for radio communications. Satellites at low inclinations with ground stations at high latitudes will be unable to communicate. As a rule of thumb, satellites must be at inclinations higher than the latitude of their ground station.

#### 2.3.6.11 Policy, Launch, and Licensing

It would take thousands of pages to explain all the policy, licensing, and launch and space safety concerns that exist when building satellites, some key factors that must be considered early in design are frequency, imager, and laser licensing and coordination, deorbit constraints, environmental testing, and launch and space hazards. The big challenge in licensing and coordination is radio frequency. The process is regularly changing, complicated, somewhat inconsistent, and dependent on a wide variety of factors. UNP provides additional guidance on frequency licensing and coordination in the form of presentations so it will not be addressed in detail here. However, it should never be ignored as failure to receive frequency licensing can, and has, caused fully functioning spacecraft on the cusp of launch to be removed from launch vehicles, forbidden from flying. Imager and laser licensing, if applicable, require some effort as well but are much more straightforward than frequency. Deorbit constraints were already mentioned in orbit design, but it should be noted that the laws governing orbital debris and deorbit timelines are in flux, so understanding current policy is important. Environmental testing and launch and space hazards are somewhat related in that this testing ensures the spacecraft can both survive the harsh environments it will be exposed to during launch and in orbit and do no harm to the launch vehicle or other ride sharing spacecraft. These environmental tests vary, but for CubeSats typically include at least a vibration test and a thermal-vacuum test. Other launch and space hazards which should be avoided, if possible, include pyrotechnic devices, radioactive devices, pressure vessels, flammable and/or caustic fluids, and unrestrained deployable systems.

**Note**

While the mission design process may feel complicated, difficult, frustrating, and messy, it ends with a complete vision for a satellite. This process is incredibly important as it limits the chaos of unknowns later in development when they are harder to resolve. When everyone agrees, the budgets align, and a meaningful mission exists, it is very exciting!

## 2.4 Detailed Design and Implementation

Detailed design and implementation are closely related. After the mission science is understood and requirements are developed, it is time to begin hardware and software development. Note that much of the high-level design occurred during requirements development. The analyses performed should not be invalid, they just need more detail. Commercial off the shelf (COTS) parts may already be determined via trade studies, or the trades may be created or revisited as understanding increases. In-house component designs, possibly prototypes, will be made based on the requirements and analyses. These early implementations will have issues, and the detailed design and implementation will iterate until a fully functioning component is realized. This section provides an overview of structural, avionics and software development.

### 2.4.1 Commercial Off the Shelf (COTS) Parts

A wide variety of subsystems exist in the commercial market and most teams will purchase most of their components from CubeSat parts vendors. This primarily applies to avionics, as they make up most of the satellite. This is discussed more thoroughly in the avionics section, but a general piece of advice applies. The quality and maturity of COTS components and vendors varies. Structures tend to be quite mature and available but are also relatively easy to develop in-house. Software quality and maturity varies widely and open-source options from NASA and others are commonly used. Avionics quality varies greatly by company, subsystem, and time as many of these companies are quite small and one or two knowledgeable engineers joining or leaving may make a drastic difference. Learn as much as you can about the company and hardware you want to use. Information sheets may be inaccurate or misrepresent the maturity of a system. Issues will occur, even with COTS parts. The quality and responsiveness of customer service should factor into your purchase decision.

### 2.4.2 Trade Studies

Trade studies involve comparing important metrics across a variety of competing options to find the best choice for your system. For example, two CDH systems may have similar inputs and outputs and roughly the same physical size and mounting pattern such that both would work in your satellite. On almost all metrics, there is no preference or benefit to choosing one over the other, but one has a much more powerful processor, significantly more storage, a higher power draw, and costs twice as much. These metrics (cost, processor performance, and storage) should be compared. Weights may be assigned to importance of various parameters, such as a team with a low budget putting a high weight on cost. Each component is ranked on each metric. The ranking system and ranges chosen should allow meaningful comparison without overcomplicating or oversimplifying. Weights are then multiplied by ranks, and each component's total score is determined. As in all engineering, the study does not replace thought, analysis, and good decision making. Just because one component has a higher score than another does not necessarily mean that component is better. It is possible a critical metric was not included, weights didn't reflect reality, or other misleading effects occurred. Even without a scored answer, a trade study is an organized way to collect information on competing products and understand key differences. This is highly beneficial when making design decisions. In addition to performance metrics, programmatic considerations may also be beneficial. These could include cost, company/customer service support, and flight heritage (how much has this component flown and how well did it work?). While it's impossible to list all considerations applicable to all satellites and subsystems, check the interfaces carefully. The best performing component in the world won't matter if it doesn't fit, can't take the right voltage or communication protocol, or causes excessive problems for other subsystems.

### 2.4.3 Structural Development

Figure 2-4 below illustrates the structural development path recommended by UNP. As always, the first step is to understand the mission and requirements. Next, a rough computer-aided design (CAD) model is generated of the structure and subsystems. As model fidelity increases, a physical model should be built. Manufacture this model to allow for easy modification; 3D printing in plastic recommended. Certain design issues, such as assembly steps, tool or hand access, and cabling designs are much more easily addressed while performing a mock assembly of real hardware than in CAD. Don't forget to design necessary mechanical ground support equipment (MGSE).

The lessons learned from the model design, assembly, and integration process are used to update the CAD and assembly procedures, then engineering model (EM) hardware is manufactured. The full spacecraft is assembled, system-level integration issues are identified, and solutions are devised. The EM cables and harnessing are also designed and manufactured at this stage. Finally, lessons learned are fed back into the CAD model, and the flight unit is machined.

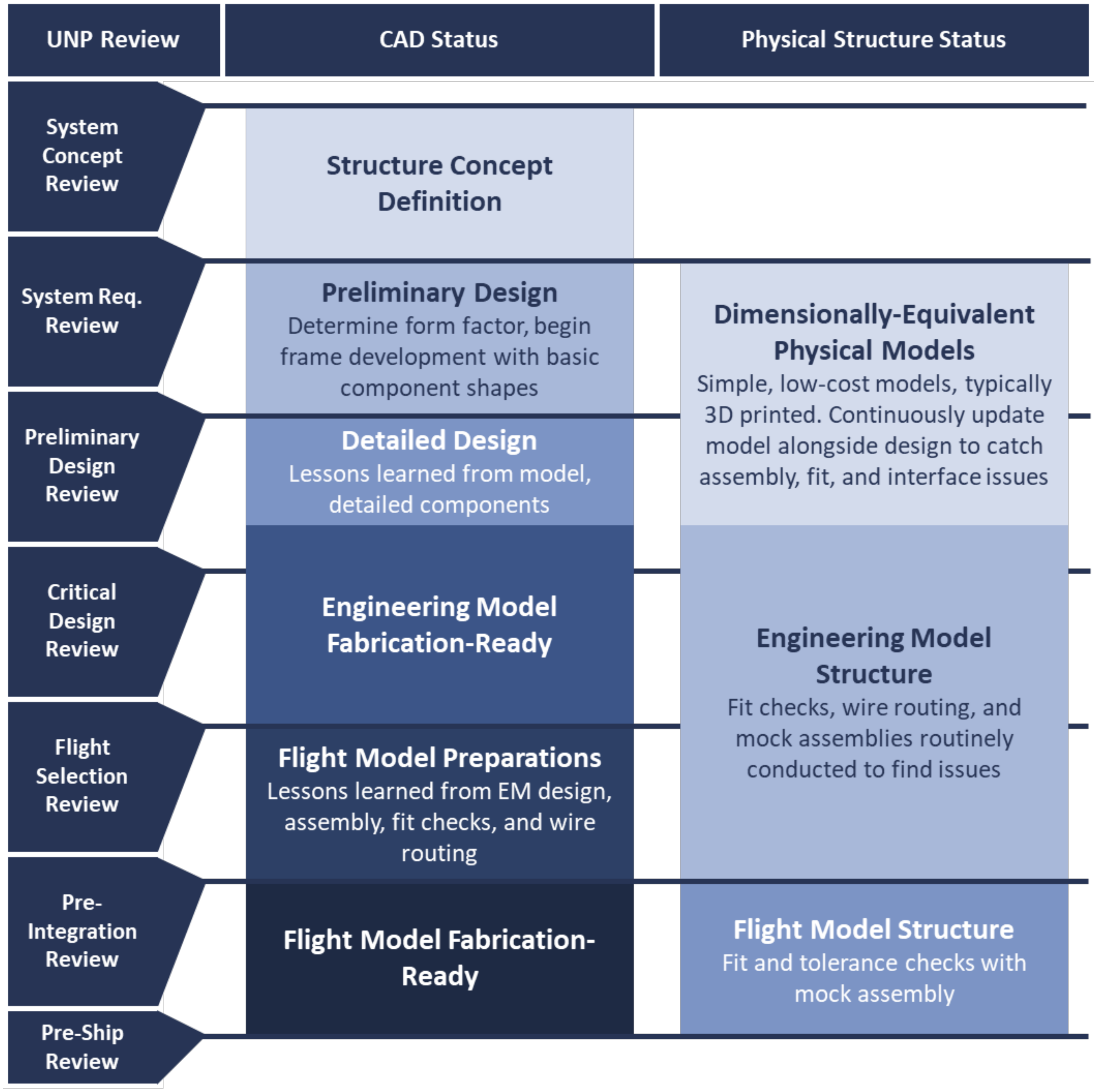


Figure 2-4: Mechanical Development Process

### 2.4.4 Avionics Development

Satellite avionics include the electrical hardware and software that support the mission. Examples of avionics units include a command and data handling (CDH) motherboard, torque rod, reaction wheel, payload interface printed circuit board (PCB), battery, or electrical power subsystem (EPS) regulation/distribution PCB. Examples of subsystems include the ADCS, EPS, CDH, or communications subsystem (COM). The system refers to all the avionics that make up the satellite.

There are many Commercial off the shelf (COTS) avionics options in the CubeSat market. Once requirements are determined, trade studies should be performed to determine whether components should be developed in-house or purchased. Take a judicious approach to in-house subsystem development. Ensure the need and level of difficulty is thoroughly understood. It is common to make simple PCBs for switching, routing, and interfacing on the satellite and the electrical ground support equipment (EGSE). Many payloads are also designed in-house. Regardless of the approach, the component must meet requirements, fulfill all necessary functions, and interface properly with the rest of the satellite.

Whether a component is made or bought, something that should be considered for all hardware is the sparing strategy. Most development will take place on what is known as the engineering model (EM), which is ideally separate from the flight model (FM). This allows the flight components to be handled with extra care, ensuring they remain in like-new condition for flight. However, limited university budgets typically mean that only cheaper components have a spare, while more expensive components are used for both development and flight. In these cases, extra care must be taken when working with the flight unit, although all components should be treated carefully including logging use and observing ESD precautions. Determine early which components will have spares and which will not. Sometimes cheaper stand-ins can be purchased or designed to cover most testing and development, only including the flight component when system functionality is mostly verified. In addition, manufacturers will regularly upgrade products or replace their product line with new versions. If a team purchases an engineering model early in development, then waits multiple years to purchase a flight model, the components may be substantially different or incompatible. To avoid rework late in development, multiple components of the same type should be purchased together. Finally, purchasing multiples greatly supports early software development. Most teams that fail to achieve flight do so due to incomplete software. Prioritizing software development from the start may be one of the most important choices a team can make and is worth additional cost.

#### 2.4.4.1 Commercial off the Shelf (COTS) Avionics

If purchasing components, perform a thorough trade study. Some components may have limited options, while others will have robust competition. Gain as much information as possible on the systems to ensure hardware and software interfaces, voltage, current, and inrush behaviors, mass, volume, and form factor, and any other necessary parameters work for your design. Be aware that the CubeSat component market is relatively immature. While some companies provide excellent products with consistent quality control, thorough documentation, and responsive customer service, many do not. Just because something is purchased from a manufacturer, does not mean it will perform as expected. Learn as much as you can about other team’s experiences with specific companies or components before deciding to purchase.

Once a component has been determined, keep the following considerations in mind. Ensure the lead time fits with your schedule, it is not uncommon for components to have lead times of 6+ months. Consider purchasing multiple at the same time, this will ensure the flight and engineering model, and any other spares in case of damage, are identical. Once received, carefully review the documentation provided by the manufacturer. The hardware should be supplied with an end item data package (EIDP). This should describe tests performed, performance observed, and any acceptance test you should perform. Think critically about the information provided and the results of your acceptance test, and determine where into the test flow (unit, subsystem, or system-level testing) the component should enter.

#### 2.4.4.2 In-House Avionics Development

If developing components in-house, expect many iterations. As avionics units are integrated and tested at the unit-level, subsystem-level, and system-level, design flaws are discovered, issues are resolved, interfaces change, functionality is modified, and team members change. Ideally iterations are limited to the unit-level, but realistically iteration exists at all levels.

While unit, subsystem, and acceptance testing may take place in relative isolation from other subsystems, eventually the subsystems will be brought together for integrated testing. This typically takes place on something known as a flatsat, or flat satellite. Design of flatsats vary, but the defining feature is that avionics are laid out near each other on a workbench, electrically integrated with cables or flatsat circuit boards, and run as they would be in the satellite. A complete flatsat should include all components except for the structure. Flatsats allow easy access for software and electronics testing before packaging everything into the limited space of the structure. As with all testing and integration, methodically building in complexity by adding subsystems one at a time will make it easier to achieve a functional flatsat. Figure 2-5 shows the recommended progression for avionics development.

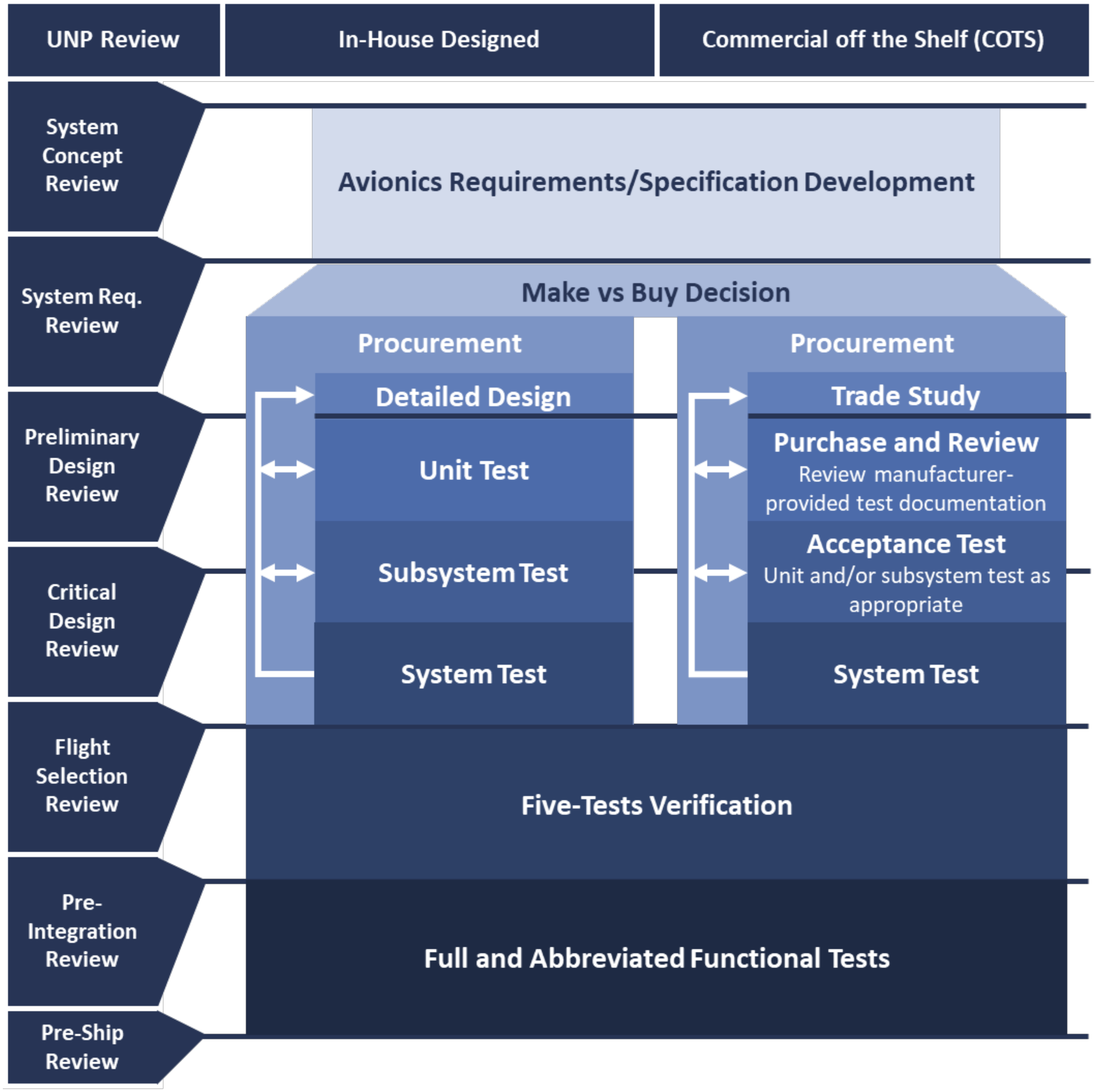


Figure 2-5: Avionics Development Process

### 2.4.5 Software Development

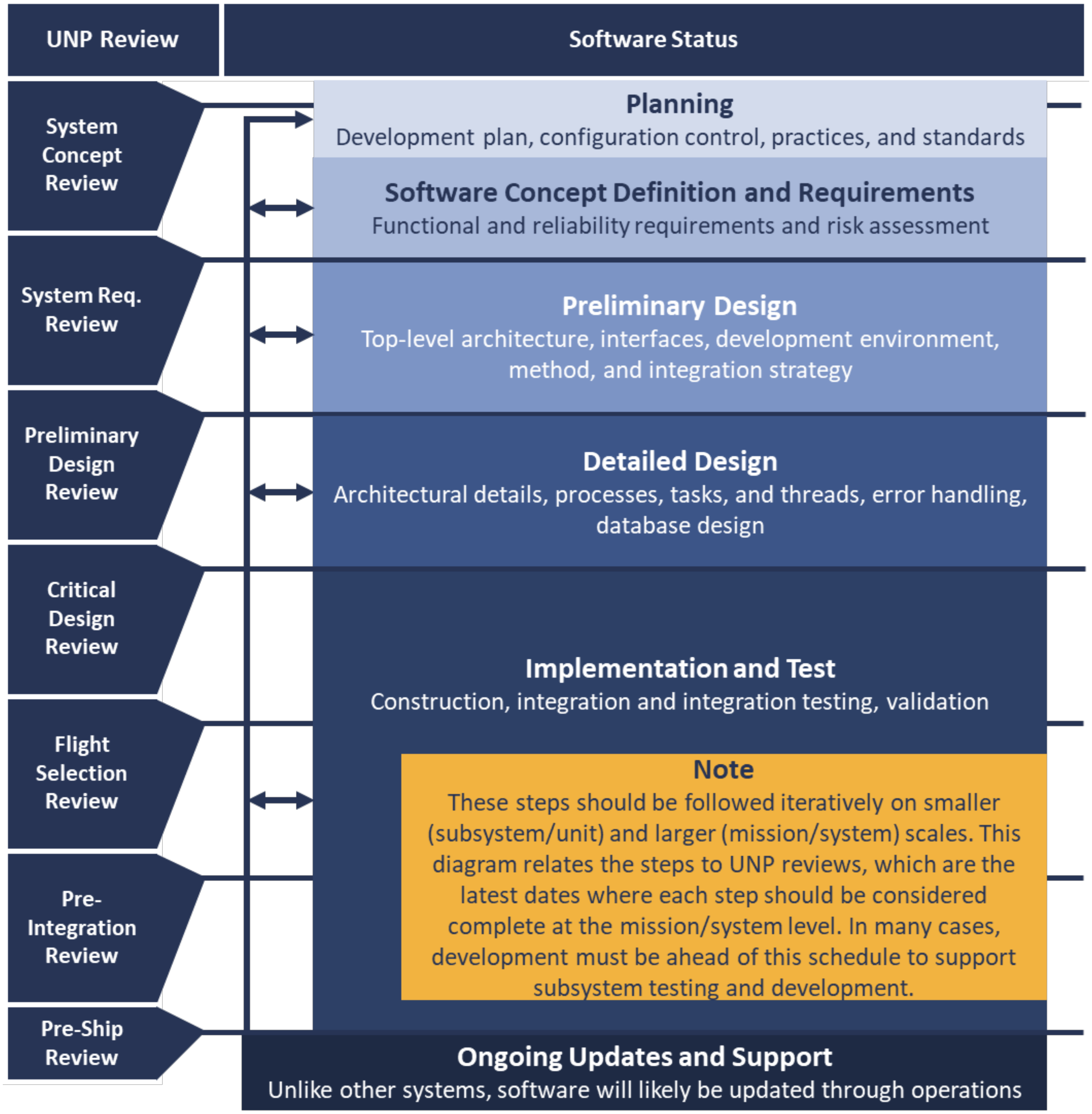
Software development may be the most difficult part of satellite development. It is not easy to track and schedule. Program managers and software team members must work closely to track software progress.

UNP expects teams to follow an iterative software development process and have robust systems in place for creation, revision, testing, documenting, debugging, change tracking, etc. The initial release creates a functional baseline that meets minimum flight worthiness. Later releases add capabilities according to priority. This approach ensures the spacecraft meets minimum flight readiness as quickly as possible. Software tends to be on the schedule critical path. Prioritizing its development and following this approach helps it stay on track with other subsystems.

Further, software is the subsystem most prone to changes as students join and leave the team. Hardware can be understood and iterated upon, but software often can be enigmatic. The bias to “Just write it myself” causes significant loss of progress. Good documentation, revision control, and training/mentoring of new team members makes a massive difference to maturity of a UNP project.

Figure 2-6 provides a rough overview of the software development cycle. Note that the steps should be followed iteratively on smaller (subsystem/unit) and larger (mission/system) scales. The diagram relates the steps to UNP reviews, which are the latest dates where this step should be considered complete at the mission/system level. Following this process ensures everything software related, including programmatic aspects, are iteratively refined. Development may not follow these exact steps; the critical piece is to ensure all steps are regularly revisited. In addition to the process shown in the figure, some additional high-level recommendations are the following:

- Choose a development method and use it consistently.
- Follow your Software Development Plan.
- Use version control.
- Use industry standard protocols.
- Consider your encryption plan early.
- Software Unit testing is an important part of the satellite development process.
- Cross-compile often.
- Add comments to your source code.
- Automate bug and error checking.



**Note**

These steps should be followed iteratively on smaller (subsystem/unit) and larger (mission/system) scales. This diagram relates the steps to UNP reviews, which are the latest dates where each step should be considered complete at the mission/system level. In many cases, development must be ahead of this schedule to support subsystem testing and development.

Figure 2-6. Software Development Process

### 2.4.6 Ground System Development

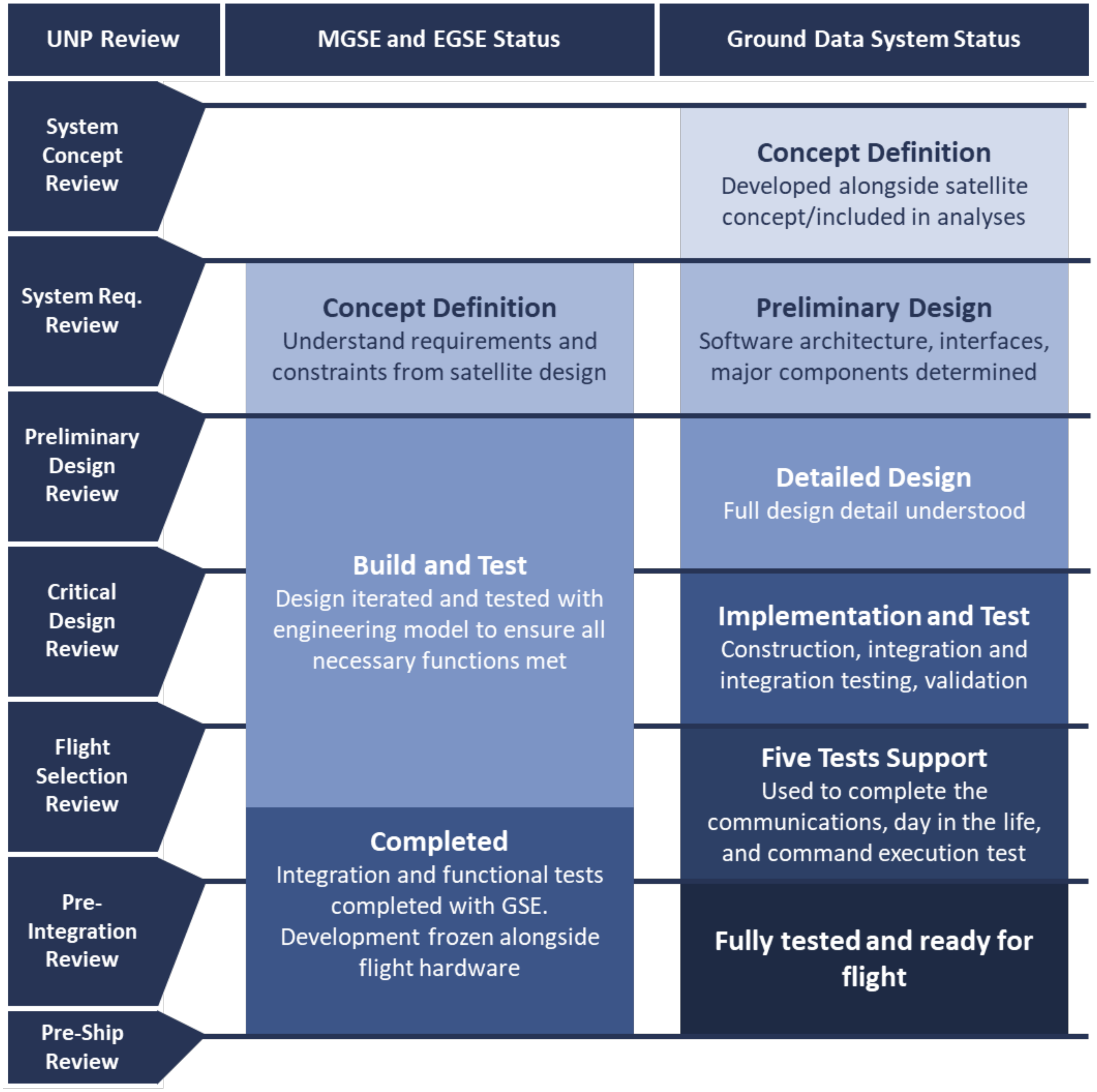
There are a wide variety of ground systems necessary for a satellite. These include the mechanical and electrical ground support equipment (MGSE and EGSE), the ground stations, and the ground data system.

EGSE stands for electrical ground support equipment. The EGSE plugs into a single connector on the exterior of the integrated satellite, allowing functions such as battery charging, software updates, and health checks. EGSE typically includes a laptop, radio, power supply, various switching and data collection systems, and the cabling to connect to the spacecraft. Since most university satellites are unique, EGSE will have unique aspects that are developed in-house. These should be developed and treated like any other software or avionics. EGSE design should be considered early to ensure both the satellite and EGSE include necessary interfaces and functionality.

MGSE stands for mechanical ground support equipment. MGSE is used to handle and protect the spacecraft during integration, testing, and shipping. MGSE may include multiple, separate components, such as a shipping case, integration stands, and a crane lift bracket for large satellites or handles for small satellites. Consider MGSE interfaces when designing the spacecraft structure.

Ground stations are the radios, antennas, and supporting hardware/software that directly communicate with the satellite over radio frequency. Ground stations must properly communicate with the satellite radio to allow command, control, and data downlink. Teams can take a variety of approaches to ground systems including purchasing a complete COTS ground station, purchasing COTS components such as a radio, amplifier(s), rotator, and antenna and building it themselves, or purchasing ground station services. Several companies, operate ground stations throughout the world. They cover a wide variety of bands and can interface with many different radios. Typically, these involve an up-front cost for interface testing, configuration, and licensing as well as ground station selection, then an ongoing cost for time used on the ground stations. Operating from multiple ground stations provides many benefits, while lack of ownership of a ground station provides some drawbacks. Ground station services are well worth considering as an alternative to ownership. Teams should consider the best path for their mission.

The ground data system includes the networks, databases, command and control software, data processing software, and other hardware/software necessary to achieve the mission and link the operators to the satellite. Most ground stations are not co-located with the operations center. Even if a team owns a ground station, it likely sits on a building roof or in a distant field. Typically control software will link to the ground station via the internet. This is also the case with ground station service providers. Not only must the command and data path between the operations center and antenna be reliable and fast, but it must also be well understood for rapid debugging and fixing during operations. To send commands and understand satellite status, software must be developed providing the necessary user interfaces. These user interfaces may be built within development environments designed for satellite operations or may be fully custom. Regardless, each satellite will have unique commands, telemetry, data, and other software interfaces to which the operations software must adapt. Finally, the ground data system will have to consider data processing and storage. All data should be stored for reference and exploitation. Raw experiment data will likely need processing on the ground to complete the mission goals. This doesn’t have to happen in the same software or on the same computer as operations but must be developed alongside to ensure the mission is achievable.



UNP Review

MGSE and EGSE Status

Ground Data System Status

System Concept Review

System Req. Review

Preliminary Design Review

Critical Design Review

Flight Selection Review

Pre-Integration Review

Pre-Ship Review

**Concept Definition**
Developed alongside satellite concept/included in analyses

**Concept Definition**
Understand requirements and constraints from satellite design

**Preliminary Design**
Software architecture, interfaces, major components determined

**Detailed Design**
Full design detail understood

**Build and Test**
Design iterated and tested with engineering model to ensure all necessary functions met

**Implementation and Test**
Construction, integration and integration testing, validation

**Five Tests Support**
Used to complete the communications, day in the life, and command execution test

**Completed**
Integration and functional tests completed with GSE. Development frozen alongside flight hardware

**Fully tested and ready for flight**

Figure 2-7. Ground Development Process

## 2.5 Assembly, Integration, and Test

As with all other development steps, assembly, integration, and test is a nonlinear process. Testing will be performed throughout design and implementation on unit and subsystem levels, with the system eventually coming together for integrated, system-level testing.

### 2.5.1 Integrated Testing – The Five Tests

Once the flatsat is functioning as expected, UNP levies five system-level tests on our teams. Successful completion of these five tests ensures the satellite has a good chance of functioning properly after launch vehicle deployment. The five tests are command execution of all commands, system-level day-in-the-life, antenna pattern and long-range communications, complete charge cycle, and ADCS functionality. Some additional detail on these tests can be found in the deliverables section of this User Guide, and full detail can be found in the UNP Nanosatellite User Guide.

### 2.5.2 Flight Assembly and Test

If the flight hardware has completed flatsat system tests, it is time for flight assembly. Numerous engineering model assemblies should have been completed at this point, ensuring robust assembly procedures and experienced integrators. Ease of assembly varies by spacecraft and is highly dependent on the consideration given to assembly during structural design. A detailed assembly procedure should not only exist but have completed multiple revisions due to engineering model development and test builds (i.e., practice this a lot and edit the procedure and components as necessary with real-life feedback). The assembly procedure should be clear enough that an inexperienced team member could complete assembly without assistance.

**Note**

Just like flight hardware development, assembly is usually performed in a clean room. An assembly fixture may also be needed. Throughout the assembly process, it is recommended to take a test-as-you-build approach. This may not be possible for every component but finding issues as early as possible is beneficial.

Once the satellite is built, it should undergo thorough functionality testing. While everything should have been tested before integration, there is always the possibility of damage, interference, or other behaviors not seen in a flatsat configuration. Testing with the integrated satellite will make heavy use of the Electrical Ground Support Equipment (EGSE) as access points on the satellite will be limited. Not all functions may be testable after assembly, it is important to test as much of the Concept of Operations (CONOPS) as possible. This process of iterative system level testing can take many months, possibly even years. While trial-filled, an organized and thorough AI&T process can reduce risk and yield a more successful mission.

### 2.5.3 Environmental Test

Environmental testing is the process of subjecting a spacecraft to simulated launch and space environments. This provides assurance that the spacecraft will do no harm to the launch vehicle or other spacecraft on the launch, and that it can survive the harsh space environment. While a wide variety of environmental test types exist, only a limited number are generally required for CubeSats. These are random vibration, bakeout, and thermal vacuum tests. Once committed to a launch, the launch vehicle provider will provide vibration levels the satellite can expect to experience, as well as slightly higher levels to which the spacecraft must be tested. The satellite must survive vibration testing in all three axes. In addition, the satellite must outgas for the required time and temperature in a bakeout chamber to ensure volatile compounds are removed. The launch vehicle provider will also require submission of paperwork ensuring the spacecraft passed the appropriate tests at the appropriate levels. Other tests and information commonly required by the launch vehicle include separation/launch vehicle deployment tests, harness checkouts, hazardous operation timer functionality, electrical inhibit diagrams and tests, and proof of frequency licensing.

**How does this relate to UNP?**

If a UNP team is ready for satellite delivery, environmental testing will be conducted at Kirtland Air Force Base at AFRL. UNP will oversee the test campaign, with students heavily involved in the process.

While the launch vehicle and other spacecraft only care about the satellite meeting do-no-harm requirements, as the satellite owners, we want to ensure it not only physically survives the launch and space environment, but also can complete its mission. This is known as mission assurance. All satellites must complete do-no-harm-testing, but we regularly do more than this to increase mission assurance (including the Five Tests). Some commonly performed tests include thermal vacuum, thermal cycling, characterization, calibration, and mission specific tests.

## 2.6 Manifest and Launch

The process of finding a launch is called the manifest process. Once a launch is agreed upon, the satellite is said to be "manifested on a launch vehicle." There are many paths a satellite can take to find a manifest. A large satellite may buy its own launch vehicle, thus securing its launch by being a primary launch vehicle payload. Small satellites are almost always secondaries, satellites hitching a ride on a rocket paid for by a primary.

There are many paths to launch for secondary satellites, some, such as UNP and CSLI provide free launches to educational satellites, while others may cost a few hundred thousand dollars. These organizations generally find launch slots on a variety of launch vehicles, match the orbital parameters with those required by the satellites, and offer rides to the satellites. If accepted, agreements will be signed providing expectations and obligations for all involved organizations. Many of the UNP-levied requirements come from standard expectations found in these agreements.

**How does this relate to UNP and CSLI?**

In UNP, a launch is sought through the Department of Defense (DoD) Space Test Program (STP). STP provides launch manifests for experiments across the entire DoD. UNP teams that progress past the down select of a Nanosatellite Program will present to the Space Experiments Review Board (SERB) each year. STP finds appropriate launches, offers them to UNP, and handles much of the interfacing with the launch vehicle provider.

In CSLI, any U.S. educational team can apply. Teams that are selected receive up to $300,000 for integration and launch services. CSLI finds appropriate launches for selectees and handles much of the interfacing between the team and the integrator/launch vehicle provider.

Due to the popularity of small satellites and ability to fit many CubeSats on a launch vehicle, most launch vehicles fly with several secondaries. Regular launches are even organized entirely for small satellites, known as rideshares. There are also many companies attempting to enter the launch vehicle market, most aimed at small launch vehicles and small satellite payloads. UNP and CSLI teams stay very aware of launch opportunities and can provide advice in this regard.

## 2.7 Mission Operations

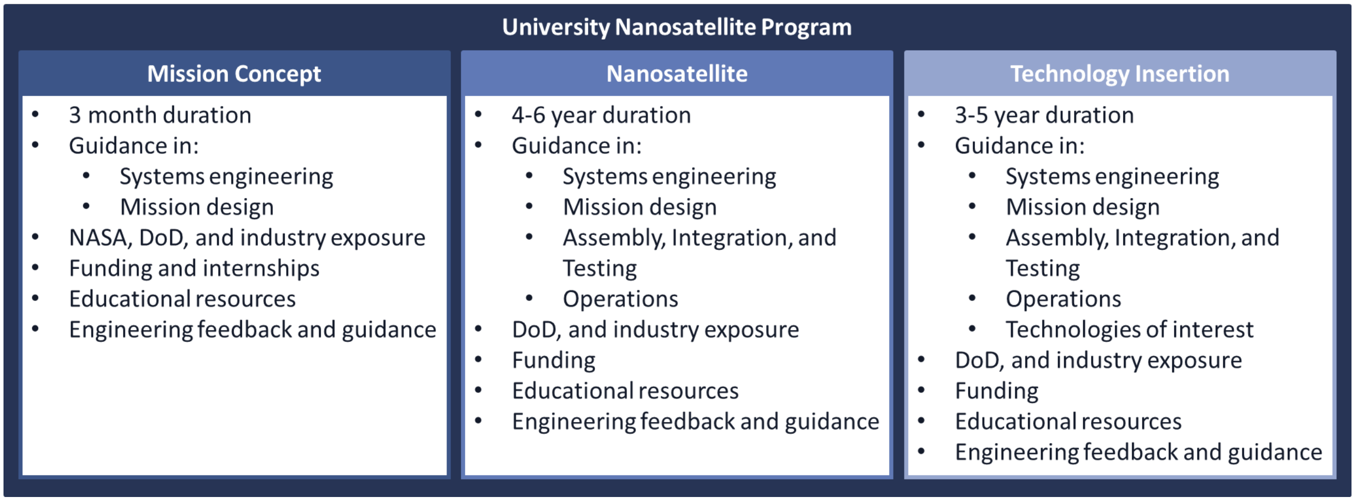
Mission operations is the most critical part of a satellite's life. While the spacecraft development process is important, a perfectly designed satellite can still fail in operations. Known spacecraft and ground system behavior, operations team personnel and experience, prior planning for nominal, and off nominal, operations, and other considerations will suddenly become very critical when the satellite reaches space and must be completed well ahead of time.

The principles of mission operations are the same regardless of satellite size or mission cost. Spacecraft safety always comes first. Organize operations in order of increasing complexity and risk, performing the safest and most minimal objectives before others. One of the biggest risks to the spacecraft is human error, with carelessness and lack of communication the most common forms.

The operations team structure, communication paths, decision making processes, and quality assurance must be carefully considered.

## 2.8 The UNP Approach

UNP runs multiple, distinct, educational program options, including Mission Concept Programs, Nanosatellite Programs, and Technology Insertion Programs. These aid university teams and the Government in various ways. The core of UNP is the Nanosatellite Program. These involve the complete design and build of a satellite. Mission Concept Programs are modeled on the first few months of the Nanosatellite Program, providing entry-level guidance to teams learning to develop small satellite missions. Technology Insertion Programs follow a similar flow to Nanosatellite Programs, but rather than university determined concepts, the Government provides a specific technology development focus. This User Guide is specific to the Mission Concept Program.



University Nanosatellite Program

| Mission Concept | Nanosatellite | Technology Insertion |
|---|---|---|
| • 3 month duration<br>• Guidance in:<br>  • Systems engineering<br>  • Mission design<br>• NASA, DoD, and industry exposure<br>• Funding and internships<br>• Educational resources<br>• Engineering feedback and guidance | • 4-6 year duration<br>• Guidance in:<br>  • Systems engineering<br>  • Mission design<br>  • Assembly, Integration, and Testing<br>  • Operations<br>• DoD, and industry exposure<br>• Funding<br>• Educational resources<br>• Engineering feedback and guidance | • 3-5 year duration<br>• Guidance in:<br>  • Systems engineering<br>  • Mission design<br>  • Assembly, Integration, and Testing<br>  • Operations<br>  • Technologies of interest<br>• DoD, and industry exposure<br>• Funding<br>• Educational resources<br>• Engineering feedback and guidance |

Figure 2-8. University Nanosatellite Program Options

The development flow in UNP generally follows the review schedule shown in Figure 2-9. Teams begin with concept and feasibility studies, leading to requirements development and detailed design. Mission Concept Programs end here, with a well-scoped and understood mission, and many requirements determined. Nanosatellite and Technology Insertion Programs will continue through detailed design reviews, and usually include a down-select review from 10 universities to 1-4. Teams that progress beyond the down-select will receive additional funding, a launch manifest, and UNP guidance through the end of life of the satellite. The four phases shown in the figure below, Design, Integration and Test, Environmental Test, and Operations are also known as Phase A, B, C, and D respectively in Nanosatellite Programs.

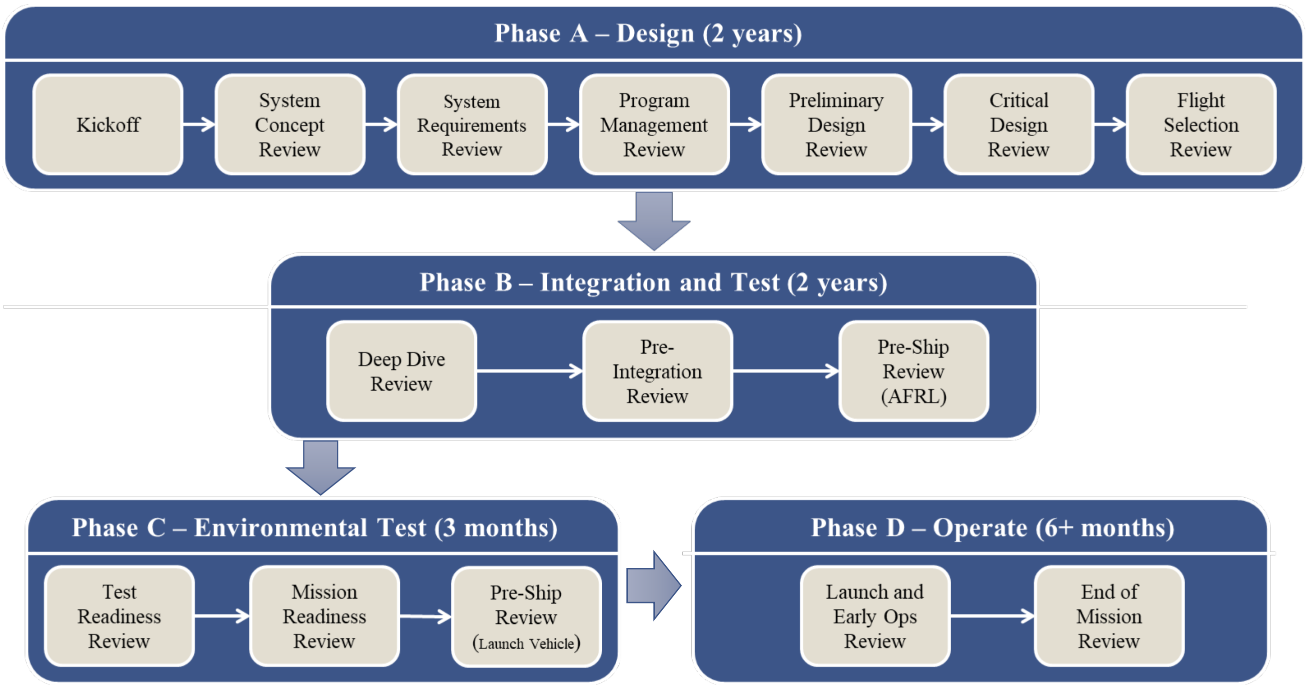


Figure 2-9. Common University Nanosatellite Program Reviews

## 2.9 Additional Considerations and Recommendations

### 2.9.1 Systems Engineering Revisited

While systems engineering was introduced at the beginning of this chapter, there are some key aspects that were not addressed. These pieces, included in the UNP-version of the systems engineering V, include the systems engineering management plan, technology transfer, and the concepts of verification and validation.

### 2.9.1.1 Systems Engineering Management Plan (SEMP)

The SEMP defines the engineering approach taken by the team. It includes details such as organizational structure and roles, personnel recruitment, training, and management, risk management approach, systems engineering process, documentation process, and many other details.

### 2.9.1.2 Technology Transfer

Anyone providing funding for your satellite likely wants something in return. In the case of UNP, highly educated students are the primary goal, but UNP also desires technology development. The mission stakeholder, providing the meaning behind the mission, may be expecting a technology demonstration and the resulting data. Providing technology development and demonstration information to others is known as technology transfer. Typical recipients of the technology transfer include Government agencies, companies, and the greater scientific community.

### 2.9.1.3 Verification and Validation

In systems engineering, these terms are not used interchangeably. Verification refers to system evaluation such as test, analysis, demonstration, or inspection to ensure it meets requirements. Validation refers to ensuring the stakeholders are pleased with the system. One common way this is phrased is verification ensures you built the system right; validation ensures you built the right system.

### 2.9.2 Configuration Management and Tracking

Another process that must be used during satellite design is configuration management. Configuration management is the way program documents are controlled and protected from modification or loss. Some questions to consider when developing a configuration management plan include the following.

#### 2.9.2.1 Documents to Control

Not every document needs to be controlled. For example, as-run test notes are important to save, but will never be revised or require formal release approval. Documents that need configuration management are system level, design, and flight hardware documents such as the Mission Design Document, RVM, budgets, CAD, schematics, and procedures.

#### 2.9.2.2 When to Control Documents

Documents will start in a draft phase, rapidly moving through several revisions. Once a document is considered ready for official release, configuration management should begin for that document.

#### 2.9.2.3 Where to Store Documents

Individual computers are the worst place to store documents due to high student turnover and general lack of backups. Some teams may also have security requirements not met by individual computers. Teams should have centralized locations for document storage to ensure data integrity. For general documentation, a well-organized and managed networked file system through the school, or possibly online such as Google Drive, may be adequate. For certain things, like software, dedicated tools such as Git should be used.

#### 2.9.2.4 Document Access

Document access should be limited to the team.

#### 2.9.2.5 Document Modification

Documents should not be modified without proper approval or process. For example, if a subsystem lead singlehandedly changes one of their requirements in the RVM, it could result in unforeseen side effects such as interface mismatches. It is recommended that each document have an owner, and only that owner may modify the document. Others must request changes from that person.

#### 2.9.2.6 Document Tracking

At a minimum, document owners should be tracked so team members know who to contact with changes or questions. There should also be document naming and filing systems. For example, all ADCS subsystem documents might start with ADCS. ADCS-TR-00 might mean “Attitude Determination and Control System, Test Report #.”

#### 2.9.2.7 Design Change Approval Process

Early in the design process, subsystem leads may be able to approve changes due to the immaturity of the system. Eventually the design will be frozen, and a process must exist for change approval. This could include a team meeting, a review with the team leadership, or other appropriate measures. All changes must be properly documented.

#### 2.9.2.8 Document Signatures

Most formal documents require various signatures before release. Signing a document means the signer has read, agrees with, and approves of the document. Documents should be signed by the author as well as the approval authority. The level of approval authority should correlate with the level and potential impact of the document to systems, subsystems, and the mission. The purpose of the signature process is to promote accountability in documentation development. A document should only be signed if the signer has read and concurs with the content.

### 2.9.3 Quality Assurance

Quality Assurance (QA) ensures your satellite is properly handled, built, and tested. Humans are hazardous to a satellite's health, such as inexperienced team members and students operating on minimal sleep. The largest rule breakers are new team members with limited knowledge of rules, and team leaders who think they are above quality assurance. Quality assurance applies to everyone. The following practices are recommended to minimize human error.

#### 2.9.3.1 Personnel Training

New team members must be made aware of rules, policies, and procedures, as well as trained on the specific hardware or software with which they will work.

#### 2.9.3.2 Two-Person Rule

No one is allowed to work on important hardware alone. What constitutes important hardware may vary by university, but generally includes all flight hardware and most engineering hardware. For example, an integrated EPS should not be worked on alone, but a cheap, unmodified Raspberry Pi used for software development could be provided to each relevant student for ease of development. This rule is commonly broken by leadership members, who consider themselves experienced enough, or those desperately working to catch up to a schedule. Assuming oneself too competent to make a mistake commonly leads to carelessness, while desperation reduces attention to detail and leads to non-methodical approaches.

#### 2.9.3.3 Mandatory Inspection Points

When a new piece of hardware arrives, it should be immediately inspected then acceptance tested. This ensures it is working as expected before further testing or integration. Components and subsystems should also be inspected and functional tested before integrated testing. In addition, for a component, subsystem, or system to be considered complete, it must be tested and verified. For your satellite to be considered complete, it must pass the pre-ship review. These are all mandatory inspection points.

#### 2.9.3.4 Controlled Access to Hardware

Hardware access should be limited. Flight hardware should be kept in locked areas, access only provided to properly trained team members. During testing, lab access should be limited to necessary personnel as it requires focus.

#### 2.9.3.5 Safe Handling Procedures

Guidelines and procedures should exist for lifting and handling components and the satellite. Team members must be aware of team policies on cleanliness and electrostatic discharge (ESD).

#### 2.9.3.6 Informing Leadership

Testers must keep leadership informed of ongoing tests.

| The Case of the “Failed” Test | |
|---|---|
| **Situation:** | A team lead was running a test. The system was idle and required little active supervision, so the tester stepped away to use the restroom, leaving the system running. While gone, another student came into the lab. The student noticed the power supply was on and turned it off, causing the test to fail. The second student promptly left, so when the tester returned, they did not realize the power supply had been manually turned off. The team lead assumed the test had failed. |
| **Problem:** | The first problem was that a test was left running with no supervision. Satellites and satellite components should be under constant supervision when active. The second problem was that the lab had no notification system for those needing to share the space. While other team members may access the lab during a test, there should be a way to notify them about ongoing operations.<br>Since the tester did not realize the power supply was manually switched off, they spent a week trying to debug the test failure, leading to development delays. |
| **Solution:** | At the team’s weekly meeting, the tester described the course of events leading to the assumed failure. The student who switched off the supply realized they had caused it and explained what occurred. Realizing this event was a problem, the team implemented the two-person rule, requiring two people to be present for all testing and work on hardware, and allowing one to step away for brief periods of time during non-critical moments. The team also developed a simple system to notify others about ongoing tests. A clip was attached to the clean tent and would hold a red piece of paper during testing. |

#### 2.9.3.7 Log Everything

If something happens to a piece of hardware, it should be logged. This is especially true for flight hardware and hardware with expirations. For example, batteries have shelf lives and connectors often have a limited number of mate/de-mate cycles. In addition to these items log entries should be made for tests, hours of use, and the last person who used it. Procedures are useful for notating what occurred, but test logs add additional benefit by compiling hardware history in one place. A large cause of hardware problems is lack of historical knowledge. Has a piece of flight hardware been over-tested? Has a specific test already been performed? Is the hardware approaching its shelf life? If a test isn’t documented, it may as well not have happened. We also strongly recommend taking pictures of hardware throughout development.

#### 2.9.3.8 Shipping Procedures/Equipment

If you are shipping or receiving a piece of hardware, consider the following. If shipping, ensure the hardware is thoroughly protected. Also, ensure the receivers notify you upon receipt and understand proper handling, including acceptance testing. When receiving hardware, apply the same principles. Does the box or shipping container look unharmed? Does the hardware look undamaged? Were you provided an acceptance test to run to ensure that the hardware is operational?

| How does this relate to UNP? |
|---|
| This section contained a lot of recommendations on management, tracking, and documentation. The goal is not to get bogged down by excessive programmatic requirements, but to ensure program integrity and knowledge are maintained across student turnover. UNP strives to implement the minimum necessary programmatic burden on our teams. If a recommended practice is excessive for your situation, consider a different approach. However, these guidelines are based on over 20 years of experience with student teams. Failure to maintain knowledge has ended more UNP missions than any other cause, so think carefully before taking a different path. |

# 3 Mission Concept Program Information

The goal of the Mission Concept Program is to teach teams to develop a mission which is meaningful to the stakeholders, well scoped to the capability of a student team, feasible to complete, and follows good systems engineering practices. Developing missions with these traits not only benefits the existing university programs, but also allows universities to submit competitive technical proposals to future CSLI, UNP, and other RFPs. Students and professors in this program will receive extensive educational materials, participate in presentations, reviews, hands on courses, and workshops, and have access to engineers with many years of experience in designing, building, testing, and operating small satellites. This chapter will cover the resources, events, and other programmatic details of the Mission Concept Program.

## 3.1 Events

A variety of educational events will occur throughout the summer. A potential program schedule can be found at the end of this chapter.

### 3.1.1 Kickoff and Mission Design Course

The program formally begins with Kickoff and the mission design course. This is a multi-day event hosted by CSLI, ER&T, and UNP. The first day includes programmatic information and the introduction to the mission design course, the middle days are workdays for the course, and the final day is for course presentations. Optional events such as tours may occur throughout the official events.

The Mission Design Course covers the UNP and SSP philosophy of rapid concept-to-flight satellite design. This approach is described in the User Guide and other UNP resources, however, applying it can be difficult without hands on experience. In this course, students practice the techniques and critical thinking exercises necessary to apply the UNP process to any mission design. Course instructors may include engineers from UNP, CSLI, ER&T, and other NASA, DoD, and Industry organizations.

### 3.1.2 Internships

Multiple students from each team will be hired as interns with Space Dynamics Laboratory. These students will live in Albuquerque for most of the internship. Throughout the summer, the interns and other participating students at the university will participate in educational workshops and exercises and will be seated near small satellite experts for constant feedback and guidance.

### 3.1.3 Workshops

Workshops will be regularly offered throughout the Mission Concept Program. These will include presentations on various small satellite topics, as well as hands-on exercises to support development of each team’s satellite. Details on each workshop will be provided as available. The tentative schedule can be seen at the end of this chapter, but it is expected that workshop specifics will be tuned to the needs of the student teams each week.

### 3.1.4 Reviews

Design reviews are a significant educational aspect of UNP. Teams will present their work, and reviewers from NASA, DoD, and Industry with extensive experience in small satellite development will provide feedback, guidance, and constructive criticism. While reviews may feel harsh, they are not intended to demoralize or levy personal attacks on students. The purpose of reviews is to expose points of improvement, so most feedback will be critique. Successful teams are open and honest about their progress, recognize reviews as an opportunity for mentorship and feedback, and rapidly implement suggestions. Attempting to distract from unknowns or issues is highly discouraged, as you cannot receive useful feedback or suggestions if issues are not exposed.

#### 3.1.4.1 System Concept Review

At the System Concept Review (SCR), reviewers will focus on the mission objectives, experiment plan, feasibility, and systems engineering approach. Most mission concepts will be modified at SCR, while some may experience significant changes. Examples of reviewer questions include the following:

- What is the experiment you want to conduct?
- What data product will be delivered from this mission?
- Why are you interested in this experiment?
- Who is your customer?
- Can this experiment realistically be done on-orbit?
- Is there value in the science of this mission?
- Has the experiment been done before?
- Why does this experiment need to be done in space?
- What is the relevance/contribution?

SCR will occur during the month of June. Interning students will present in person, while the rest of the university team will attend virtually. Review dates and times will be scheduled in early June and presentation content expectations will be provided. SCR will last for a maximum of two hours: 45 minutes presentation and the rest for discussion.

#### 3.1.4.2 System Requirements Review

At the System Requirements Review (SRR), reviewers will focus on requirements at all levels, as well as the team's approach to requirements management. Reviewers will scrutinize the team's requirements verification matrix (RVM) and provide suggestions for new and/or clearer requirements. In addition, it is an opportunity for teams to demonstrate proper systems engineering flow down. Writing requirements takes practice and the RVM is expected to experience many revisions. SRR is an opportunity for extensive feedback on this topic.

SRR will occur during the month of July. Students will not make a formal presentation; SRR is discussion-based.

### 3.1.5 Closing

In late July, students and PIs will attend a closing event hosted by UNP in Albuquerque, NM. Teams will present their work to a review board and receive verbal final feedback. In addition to the usual reviewers, this event is likely to include senior leadership from the Air Force Research Labs and other Government entities. This feedback generally includes advice on mission relevance to the DoD and NASA which is very important for developing meaningful missions. UNP will also provide guidance on future opportunities, next steps, and closing comments.

Additional activities associated with the closing event may include tours of Air Force Research Lab Space Vehicles Directorate facilities and presentations from AFRL senior leadership.

### 3.1.6 Small Satellite Conference

Small Satellite Conference (SmallSat) is hosted by Utah State University and managed by Space Dynamics Laboratory annually in August. This is the largest conference in the field of small satellites and includes presentations, industry booths, student recruiting events, and much more. There is no better way to immerse yourself in the field of small satellites than attending SmallSat. Teams are highly encouraged to attend with as many students as possible. The conference includes many student-focused events including networking mealtimes, research paper and poster competitions, and student booths. Student registration and other student-focused fees are quite affordable, though housing costs can be high. Housing should be reserved as early as possible (6+ months), as the conference overwhelms the small town of Logan, UT. The funding provided to teams participating in the Mission Concept Program could be used to send as many students as possible to SmallSat. Additional details can be found on the SmallSat website in the resources section.

## 3.2 Provided Resources

UNP provides extensive educational resources to teams which are described below.

### 3.2.1 Presentation Slides

All presentation slide files will be provided to the teams for reference. This includes kickoff, mission design, workshops, and other existing UNP resources.

### 3.2.2 User Guide

In addition to this Mission Concept User Guide, UNP develops a Nanosatellite User Guide. With further guidance on all aspects of small satellite development. Portions of that guide, such as the requirements list, are included in this guide for reference. Due to DoD limitations on information release, the Nanosatellite User Guide is currently only available to teams participating in Nanosatellite Programs. UNP expects this guide to become publicly available soon. All publicly released documents are available on the resources tab of the UNP website.

### 3.2.3 Expert Area Telecon (EAT) Recordings

UNP holds a series of 25+ educational presentations with all Nanosatellite Program participants. The series begins with systems engineering and other high-level topics, eventually covering all subsystems and many supporting topics. The presentations most applicable to mission concepting will be provided to the teams at kickoff, with additional presentations made available as they become publicly releasable. Recordings of these presentations can be found on the resources tab of the UNP website.

### 3.2.4 Engineering Support

One of the greatest benefits of participating in UNP is access to expertise! Due to our position in AFRL and extensive history with numerous universities, industries, and alumni, UNP has access to expertise across all aspects of small satellites. If the UNP, CSLI, and ER&T teams can't answer your questions, we will find someone who can. During the internships in Albuquerque, small satellite experts will be next door for immediate access.

## 3.3 Schedule

The tentative schedule for the Mission Concept Program is shown below. This is subject to change. Details and changes will be provided as early as available. All events will be available via virtual meeting software such as Zoom. Interning students are expected to attend events in-person, and remote personnel are highly encouraged to attend virtually. All student teams are welcome to attend all events, including reviews. While it is not required that teams attend reviews of other teams, students are encouraged to attend at least one or two additional reviews besides their own. This will expose them to a broader array of topics and feedback, providing educational benefit. Student expectations on information to prepare and present will be provided ahead of each event.

Table 3-1. Potential Program Schedule

| Week | Events |
|---|---|
| **Week 1:** | |
| | Intern Orientation |
| **Week 2:** | |
| | Kickoff at Kennedy Space Center, Florida |
| **Week 3:** | |
| | Instructional videos available for review |
| **Week 4:** | |
| | Interns first week in Albuquerque, NM |
| | Workshops: Mission Design & CONOPS, Relevance, and Team Management |
| | University Mission Design Presentations |
| **Week 5:** | |
| | Workshops: System Concept Review (SCR) Expectations, Power Subsystem, and CDH/SW Subsystem |
| **Week 6:** | |
| | Workshops: Communications and ADCS Subsystems |
| **Week 7:** | |
| | Workshops: Structures and Thermal Subsystems |
| | System Concept Review |
| | Space Dynamics Laboratory Intern Event |
| **Week 8:** | |
| | Workshops: Facilities, Ground Ops |
| **Week 9:** | |
| | Workshops: Satellite Ops, F' Flight Software |
| **Week 10+:** | |
| | System Requirements Review |
| | Final Event in Albuquerque, NM |
| **Week in August:** | |
| | Small Satellite Conference |

## 3.4 Other

As Mission Concept Program continues to impact various universities and students, we would love your feedback! Formal feedback calls will occur throughout the program, but we always welcome suggestions for improvement. Please reach out to any UNP, CSLI, or ER&T personnel with feedback.

# 4 Required Documentation

## 4.1 Why are there required deliverables?

Regardless of development status, documentation is a critical aspect of a successful program. Documentation provides continuity across the satellite development process and ensures everyone involved can understand the mission and status. This chapter describes documentation UNP participants are required to submit throughout the program. Templates are not provided for most required documents. Teams are encouraged to develop templates for their teams to provide consistency, and to follow practices provided by UNP.

| The Case of the Problematic Payload | |
|---|---|
| **Problem** | A university was developing a spectrometer instrument for their mission. The initial payload team failed to appropriately document the design and testing of the instrument. When this initial team left, the new team had limited understanding of past issues and solutions with the hardware. Due to lack of documentation, the new team had to redesign the subsystem. However, they did not learn from the prior mistakes and also failed to document their work. This cycle continued through several generations of team members, each wasting significant time and effort. |
| **Solution** | Eventually, the team learned from their mistakes and began to keep accurate designs and logbooks on file. Tests were followed by reports with as-run notes included. |
| **Result** | The team was able to successfully survive turnover through adequate documentation of design and tests. The spectrometer eventually made it to space, but substantial time and effort would have been saved through appropriate documentation. |

As the satellite matures, documents should mature alongside. UNP requires the same documents at various reviews with increased detail each submission. In addition, early design documents are still useful in later stages of development but may serve different purposes. For example, design requirements serve as a guide when designing a system. Once developed, requirements become a metrics against which system performance is measured.

UNP strives to avoid burdening teams with unnecessary documentation. The documents required are the minimum set necessary to ensure team continuity within the university as well as clear communication to UNP and other stakeholders. Teams are encouraged to create additional documents if beneficial, and UNP may also require additional deliverables on a case-by-case basis. A list of items that are not required for delivery but are highly recommended is also provided. These items will progress in maturity as time passes and milestones are met. We have provided guidance on development level of these items alongside required documentation.

While most of the included documents will not be required in the Mission Concept Program, they are included for reference. Tables at the end of this chapter show documentation required at milestone events.

## 4.2 System Level Documentation

System level documents involve the entirety of the spacecraft and its mission. These documents are critical to all aspects of the mission and will be active and applicable through the entirety of the satellite's lifetime. Changes to these documents flow down through the whole mission and often affect the satellite design; therefore, these documents must be controlled and maintained by the PM and CE. These documents require significant effort, thought, and review as they apply to the entire mission.

### 4.2.1 Mission Design Document

The first systems level document that your team should develop is the Mission Design Document. This document will answer questions such as, “Why are we building this satellite?” and “Are we meeting our original/current/documented goals?” This document has three key sections: Mission Overview, Experiment Plan, and Concept of Operations (CONOPS). These sections can be ordered or combined as appropriate to your team, but all sections must be included. Brief descriptions of each section are provided below. The Mission Design Document is required, with increasing detail, at every review.

#### 4.2.1.1 Mission Overview

This is a brief overview of the mission used to introduce new students, potential sponsors, and reviewers to the mission. In addition to introducing the mission, it provides a high-level overview of the satellite, includes mission objectives and success criteria, and discusses the mission’s stakeholder relevance.

In addition, the following five questions help define the mission. They should be answered in the mission overview section.

1. Who is your customer?
    a. A customer is a user of the mission data or a sponsor/stakeholder. They may already have expressed interest or could be those you expect would benefit. Stating “AFRL” as a customer does not capture enough detail. A name, office, or program would be examples of a particular customer.
    b. UNP does not qualify as a customer of a program’s data.
2. What are your customers’ requirements?
    a. What mission-driving requirements come from the user?
    b. Does the user care about the way the mission is performed?
    c. Does the user care about the way the data is collected?
    d. Does the user care about the type of data?
    e. Does the user care about the accuracy/precision of the requirement?
3. How will you demonstrate that you have met your customers’ needs/requirements?
4. What are the technology options?
    a. What option(s) exist for your primary payload?
    b. What other payloads or capabilities are available and what led to your selection?
5. What are the risks associated with the selected technologies?
    a. What is the Technology Readiness Level (TRL) of your payload?
    b. What are the technical performance risk drivers?
    c. Describe the schedule and cost risk drivers.

#### 4.2.1.2 Experiment Plan

The experiment plan details the planned experiments. It is not a step-by-step procedure, but rather includes what payload is being used, what it is measuring/performing, what the goal/result of the experiment is, and what data products and/or high-level vehicle operations are required. Additional details should be added, as necessary, to clarify the experiment. Some satellites may only have one experiment, others may have several.

The team should understand the experiment plan and implementation early in the design process as it will drive many aspects of system design. This section explains how the mission is achieved and informs CONOPS.

#### 4.2.1.3 Concept of Operations (CONOPS)

The CONOPS describes all phases and modes of a spacecraft, including timeline and spacecraft operations. The key to a well-developed CONOPS is to capture all aspects of the mission life, the mission timeline, possible functions (typically described within satellite modes), flight rules, and possible transitions between functions in a satellite's lifetime. CONOPS should describe phase, mode, and state entrance and exit criteria as applicable. Ensure the meaning of phase, mode, state, and other terms are thoroughly clarified. CONOPS should also include hardware states, expected operation and experiments, and ground station operations. Flow diagrams and charts are encouraged. The CONOPS outlines the process and procedure by which the fully developed satellite will fulfill its mission.

This section will grow in detail and experience many revisions as the mission matures. However, it is important for this document to begin in the early satellite development stages, as the implementation of the mission and the design of the satellite are very intertwined.

A fully developed CONOPS is the baseline for all Day in the Life (DitL) testing (the DitL should verify the CONOPS) and serve as the baseline for mission operations plans. Subsystem designs and functionalities will also have CONOPS-driven requirements. CONOPS will influence the satellite from development through mission operations and the attention given to developing this document should reflect that.

Table 4-1. Mission Design Document Expectations

| Review | Section | Level of Expected Development |
|---|---|---|
| SCR | Mission Overview | Comprehensive first draft of the document; all expected information should be known. |
| | Experiment Plan | First draft. Details specific to hardware/software selection may be TBD, but high-level needs for mission objectives and success criteria should be discussed. |
| | CONOPS | First draft. Details specific to hardware/software selection may be TBD, but high-level needs for mission objectives and success criteria should be discussed. |
| SRR | Mission Overview | Include updates and feedback from SCR. |
| | Experiment Plan | Include updates and feedback from SCR. Also include details and constraints determined during requirements development analyses. |
| | CONOPS | Include updates and feedback from SCR. Ideally consider modes (components on/off state, pointing state, entrance/exit criteria), as these may drive/be driven by requirements. |

### 4.2.2 Data Budget

The data budget should capture the data production, storage, and handling abilities, as relevant, of each subsystem. The RVM and CONOPS should clearly tie to the budget. It should capture the spacecraft's bus utilization and on-board storage needs. These needs should direct the command and data handling (CDH) design. It should also capture a preliminary analysis of data downlink rates and availability to inform the design of the communications system.

### 4.2.3 Link Budget

The link budget ensures the chosen hardware and configuration of the communications system and ground system can receive each other's transmissions. It should include transmitter powers, cable losses, antenna gains, free space loss, receiver sensitivities, data rates, modulation schemes, and margin. The RVM and CONOPS should clearly tie to the budget. Numerous examples from the small satellite and amateur radio communities can be found online.

### 4.2.4 Mass Budget

The mass budget should capture the masses of all subsystems in the satellite, including primary structure, housings, mounting brackets, fasteners, wires, connectors, and other significant mass items. Mass budgets should list mass and contingency for each satellite component, unit, and/or subsystem as appropriate. Total mass and margin should also be calculated.

All masses should have associated contingency applied at the component level to account for uncertainty. The following contingencies are recommended by UNP:

- 25% for anything that has not been fabricated.
- 10% for massed engineering model structures.
- 5% for massed flight model structures to account for bolts or harnesses that may have been overlooked.

**Note**

Contingency and margin are often confused. If contingency is 25%, a 1,000 g subsystem should have 250 g added to the mass budget for an allowed/expected subsystem mass of 1,250 kg. Margin is calculated as follows:

Total Allowable Mass – Calculated Mass = Margin.

For a 6U satellite with a current mass budget of 11 kg, the margin would be 1 kg since the total allowable mass is 12 kg.

### 4.2.5 Pointing Budget

A pointing budget determines knowledge and control needs on satellite position and attitude. It should include performance metrics for all sensors and actuators relevant to attitude and position, as well as filter/algorithm errors. The RVM and CONOPS should clearly tie to the budget. While precise control requires comparatively precise knowledge, the reverse is not true. Some missions may require both, while others may only need precise knowledge. The budget should show the expected attitude knowledge, and control error, as well as position knowledge, is smaller than required for the specified CONOPS mode.

Momentum needs and capabilities are also part of the pointing budget. This part of the budget should compare spacecraft slew rate and momentum storage capabilities with requirements. It should also show that the satellite can accomplish required maneuvers at required times including actuator saturation and other limitations.

### 4.2.6 Power and Energy Budgets

The power and energy budgets capture the power and energy needs of the spacecraft. The RVM and CONOPS should clearly tie to the budgets. Power and energy generated and available are compared to that consumed in expected modes, phases, and attitude states. As hardware is tested and CONOPS matures, power budget generations, draws, and other parameters must be updated.

The power budget ensures the generation, storage, regulation, and distribution systems can accommodate necessary power draws. Power generation should be modeled based on orbital solar input and solar cell efficiency.

The energy budget ensures real-life timelines/CONOPS won't drain the battery. For example, the satellite passes in and out of the sun every orbit, but also may transmit to the ground, run the payload, or perform other high-power draw operations. It will also enter low power generation states, such as tumbling. Modeling these inputs and outputs over time, through a variety of realistic scenarios, is the purpose of the energy budget.

Note that total power, energy, and margin should be captured in the budget. Power and energy needs of each unit/subsystem should include appropriate contingency. For trustworthy COTS components, UNP recommends a contingency of 5%, while immature, in-house designs should loosen this to the following:

- 25% for untested avionics.
- 10% for unit-level verified engineering model avionics.
- 5% for system-level verified flight model avionics.

Together, the power and energy budget allow sizing of the Electrical Power Subsystem (EPS) and associated solar panels and battery. While these components should be determined from mission needs, it is common for the power and energy budgets to influence CONOPS. Even early versions of this document must capture all subsystem needs. As the document develops, the phases, modes, and other CONOPS impacts should be included. This document is often critical to design decisions due to limited power and must be considered in all component trades.

### 4.2.7 Requirements Verification Matrix (RVM)

The Requirements Verification Matrix (RVM) includes all objectives, success criteria, and requirements. Requirements should flow down from the high-level mission statement and mission objectives to the payload and subsystem requirements.

The RVM must be kept up to date, ensuring all subsystems and team members can reference it at any time. While requirements should generally flow down, it is ok to side-load certain lower-level requirements as appropriate. UNP discourages creating false requirements at intermediate levels to create a clean flow down. Provide notes and justification to ensure everyone can understand the source, and only break proper flow down with good reason.

Development of the RVM should start early in the design process. A nearly complete RVM is necessary to inform detailed design. Later in development, the RVM shifts from a design guide to a verification standard. As each requirement is verified, the RVM should be updated with requirement status and verification document.

### 4.2.8 Schedule

The schedule should include hardware/software development timelines and consider schedule drivers such as long lead times and system/subsystem interdependencies. The critical path, the longest path through the mission's build schedule and interdependencies, should be highlighted. Since UNP sets the schedule for reviews, the critical path and schedule dependencies help the team and reviewers assess progress relative to the UNP schedule.

The schedule should grow in detail and become task based as tasks are determined. It is not adequate to say, “Subsystem X will be completed in Y amount of time.” List the quantifiable steps towards completion for each subsystem.

## 4.3 Design Documentation

Design documents cover the design of units, subsystems, or the system. Management should approve these documents and understand the contents; however, subsystem leads, and team members are often responsible for them. These documents must be tracked and checked against system requirements. At some point in development, each design document will be frozen. This means it is considered complete and final, any further changes must go through a more rigorous change process to avoid unintended side effects.

Most of these documents are not required in the Mission Concept Program. Brief descriptions are included for reference, and teams are welcome to develop them as deemed beneficial.

### 4.3.1 Block Diagrams

Block Diagrams should be clear, easy-to-read, visual representations of the subsystem or system, and capture the basic functionality and interfaces.

Block diagrams are the first design document required in the Nanosatellite Program. Block diagrams may change or grow in detail over time but should be finalized by the end of the build and development process.

### 4.3.2 Ground Data System

The Ground Data System Document includes all ground hardware and software used for command, control, and operations of the satellite. This system includes some or all the following: antenna, antenna mount, radio, and associated Radio Frequency (RF) hardware, computer, software, and networking system. Ground station development does not need to progress as quickly as the space vehicle in the design and development process, but satellite radio, antenna, pointing control, orbit, data output, data content, and power budget, as well as frequency licensing, can influence the design of the Ground Data System. The Ground Data System Document should provide an overview of the system design and functionality. It should also include operation instructions or other relevant instructions related to the system that are not covered in the Mission Operations Document.

### 4.3.3 Ground Support Design

The Ground Support Design Document describes the design of the Electrical Ground Support Equipment (EGSE) and Mechanical Ground Support Equipment (MGSE).

### 4.3.4 Interface Control Documents (ICD)

An Interface Control Document (ICD) defines the interfaces for each unit/subsystem, including mechanical, electrical, thermal, and data interfaces. ICDs capture box/board dimensions, physical mounting, pin outs, software commands/responses, information needed to integrate the unit/subsystem, and any data expected or produced by the unit/subsystem.

It is important that each subsystem lead considers the interfaces of their components and the components with which those interface. All interfaces must be documented and tracked starting early in the design process to ensure interface agreement and proper design.

### 4.3.5 Master Equipment List

A Master Equipment List (MEL) is a list of all components and parts inside the spacecraft. This includes the parts that make up PCBs and within COTS components if possible. The MEL should reference relevant vendor-provided documentation for COTS parts relevant design documentation for in-house parts. Note that this document could also be considered a Bill of Materials (BOM) and is distinct from the materials list.

### 4.3.6 Materials List

The Materials List includes all materials in the spacecraft. It captures material type, not components. Materials in small components covered with conformal coating (parts on PCBs) do not have to be accounted for in the materials list. In addition to material type, the materials list must include collectible volatile condensable material content (CVCM) and total mass loss (TML) for each material. It is also beneficial to list where in the satellite each material is found for ease of future changes. The team should consider materials during component selection to ensure outgassing requirements for CVCM and TML are met.

### 4.3.7 Software Design Document

The Software Design Document describes software design, management, and status. It should include planning, requirements, preliminary design detailed design, and testing information. This includes software architecture definition, block diagrams, hardware and software selections, fault handling approach, version control, ICDs, and the software development process including how and when code reviews occur, how unit testing is approached, roles and responsibilities of software team members, and how schedules and tasks are tracked.

### 4.3.8 Structural Analysis

The Structural Analysis shows the satellite design is adequate to withstand the launch environment without failure. The structural analysis should include a structural overview, loads and constraints, factors and margins of safety (if applicable), results, and any other necessary information.

### 4.3.9 Thermal Analysis

The Thermal Analysis shows the satellite can survive and operate in the space environment. It should quantify the heat transfer within the satellite and to and from the environment. The analysis document should include component temperature limits, thermal model assumptions and parameters, heat sources and their duty cycles, and results.

**Note**

If available, operating and survival temperatures should be provided for all components.

- Operating temperature: The temperature at which a unit will successfully function and meet all specifications.
- Survival Temperature: The temperature, if exceeded, at which the unit will suffer permanent damage.

The thermal analysis should increase in maturity over time. Initially a one-node model should be developed to provide baseline results and develop student understanding of the model. This will grow into a more complex multi-node model as precision is needed. UNP may provide a one node reference case which teams can use to check their models. Each time complexity increases, results should be compared to the prior, more basic model. While results will vary slightly, overall consistency with former results provides increased confidence in the new model. Thermal modeling should begin early in the design process and increase in detail as the satellite design matures.

## 4.4 Assembly, Integration, and Testing (AI&T) Documentation

AI&T documents involve system integration, assembly, or testing. These deliverables span from early design through delivery to the launch vehicle. Note that many of these documents are result-oriented and are dependent on test completion.

Most of these documents are not required in the Mission Concept Program. Brief descriptions are included for reference, and teams are welcome to develop them as deemed beneficial.

### 4.4.1 Assembly Procedures

Assembly Procedures provide instructions for subsystems and system assembly. Assembly procedures should be written such that an inexperienced student could properly assemble the satellite. It should capture the task-like nature of a procedure, guiding the assembler step-by-step. Pictures and diagrams are strongly encouraged. Quantities of parts, fastener torque specifications, specific assembly instructions for each part, and proper safety/and cleanliness considerations should be listed. Note that the build-up of the satellite should be captured in such a way that subassembly procedures precede full system integration and assembly.

### 4.4.2 Avionics Test Procedures and Results

Whether a satellite is made from Commercial Off the Shelf (COTS) units, in-house units, or a mixture, units/subsystems/systems must be tested individually before the Five Tests can occur. A good test procedure includes every task the test executor must perform. The test results document should capture both the as-run procedure and the results from that procedure. It is important with any test document to summarize the data, making the outcome of the test apparent. Anomalies and failures must also be included. As avionics may change over time, test results must be for the latest version. Past testing documents should never be overwritten or deleted. Historical information can be vital to understanding anomalies in updated designs.

### 4.4.3 Full Functional Test Procedure and Report

The Full Functional Test (FFT) procedure and report and Abbreviated Functional Test (AFT) procedure and report are critical as the satellite ships to AFRL for environmental testing. These procedures must outline every step required for the FFT and AFT as the test operator may not always be from the university team.

### 4.4.4 Ground Support Equipment Manual

Ground Support Equipment (GSE) encompasses both the EGSE and MGSE. The GSE aids in handling, testing, and communicating with the satellite while on the ground. Before shipping the satellite to AFRL, the team must prepare a manual outlining the use of EGSE and MGSE. The Ground Support Equipment Manual should include all mechanical and electrical interfaces, handling requirements, and step-by-step instructions for mounting the satellite to the MGSE and connecting the satellite to the EGSE.

### 4.4.5 Satellite Hardware and Software (not a required deliverable)

Physical hardware and source code software are not direct deliverables throughout UNP. It is the responsibility of each team to capture critical elements of their mission’s hardware and software development in deliverables such as the schedule, mission design document, RVM, system budgets, interface control documents, and test procedures/results.

## 4.5 Five Tests Verification

The Five Tests are UNP’s baseline for a functioning spacecraft. These system-level tests must be passed on both the EM and FM to demonstrate functionality and nominal operations. The test procedures should be detailed enough that someone with minimal familiarity can run the test. They should also include pass-fail criteria, personnel descriptions, and an equipment list. The as-run test procedure should include notes, issues, results, the test outcome, and signature/initials of the tester as appropriate. Note that the Five Tests do not capture mission-specific functions or nuances. Additional mission-specific testing is recommended. The Five Tests may be met via separate tests with overlapping coverage. Acceptable alternatives may be discussed with UNP.

Table 4-2. Five Tests Description

| Test | Brief Description: See Nanosatellite User Guide for Complete Expectations |
|---|---|
| **Day-in-the-Life** | The satellite is run through at least 24 hours of operation (Nonstop preferred and longer duration preferred) in a flight-like setting. Software modes; eclipse, communication, and experiment periods; initial deployment sequence; spacecraft interaction via radio; and other parameters are as flight-like as possible. |
| **Command Execution** | Every software command is sent to the satellite and responses are checked for validity, errors, accuracy, etc. |
| **Complete Charge Cycle** | The satellite is allowed to discharge, ensuring power-saving and battery protection modes and features activate properly, then the satellite is charged through a flight-like interface, ensuring functionality returns as expected. |
| **Simulated Communications** | The satellite antenna gain pattern is tested in a flight-like configuration, and commands are sent and received to/from the ground station. |
| **ADCS Functionality** | ADCS sensors and actuators are checked for basic functionality and proper polarity. Algorithms and behavior are vetted as thoroughly as feasible. |

## 4.6 Miscellaneous Documentation

There are several documents that do not fit into a well-described category. Rather than technical details, these cover topics such as team management, legal documentation, and status updates.

Most of these documents are not required in the Mission Concept Program. Brief descriptions are included for reference, and teams are welcome to develop them as deemed beneficial.

### 4.6.1 Dashboard

The dashboard is a quick summary of unit, subsystem and system-level hardware and software development and testing status. UNP will provide a template.

### 4.6.2 Proof of Licensing

Teams must be aware of necessary licensing. This includes, but is not limited to, Federal Communications Commission (FCC) radio frequency licensing and National Oceanic and Atmospheric Administration (NOAA) imaging licensing. License application submission should occur when the satellite's launch is known, but understanding the process and requirements is critical throughout development to ensure the system is licensable. While not a formal license, missions flying lasers must coordinate with and receive approval from the Laser Clearing House and Federal Aviation Administration (FAA).

### 4.6.3 Waivers

If the satellite design or programmatic approach is not in compliance with any UNP guideline, the university must submit a waiver request. This must capture which guideline is being addressed, why a waiver is needed, and an explanation of the proposed approach.

### 4.6.4 Protection Plan

UNP recommends each team develop and maintain a protection plan. This plan addresses security measures taken to protect the mission. While some missions may be open source, there are still aspect that must be protected including physical hardware and satellite commanding. The protection plan should include the following:

- Security of Documents and Designs: How are documentation access and editing controlled?
- Hardware Security: How are satellite and ground station hardware access controlled?
- Command Authentication: How is satellite commanding protected (encryption, personnel access, etc.)
- Downlink Protection: Is downlink protection needed? If not, can anyone with a ground station interpret satellite transmissions?

Note that UNP does not levy any Department of Defense (DoD) requirements on the design beyond standard satellite practices.

## 4.7 Mission Operations Documentation

Mission Operations Documents describe on-orbit operations, operator training, ops team structure, and other relevant details. These documents are discussed in the Nanosatellite Users Guide.

## 4.8 Deliverables Check List

### 4.8.1 Design and Development Deliverables Check List

Table 4-3 indicates when each design deliverable must be submitted to UNP. Note the Mission Concept Program only includes SCR and SRR.

Table 4-3. Design and Development Deliverable Checklist

| Deliverable | SCR | SRR | PMR | PDR | CDR | FSR |
|---|---|---|---|---|---|---|
| Assembly Procedures | | | | | • | • |
| Block Diagrams | | | • | • | • | • |
| Command and Telemetry List | | | | | • | • |
| Data Budget | | • | • | • | • | • |
| Dashboard | | | | • | • | • |
| Ground Data System | | | | | • | • |
| Ground Support Design | | | | | • | • |
| Interface Control Documents | | | | • | • | • |
| Link Budget | | • | • | • | • | • |
| Mass Budget | | • | • | • | • | • |
| Master Equipment List | | | | | • | • |
| Mission Design Document | • | • | • | • | • | • |
| Mission Operations Plan | | | | | • | • |
| Pointing Budget | | • | • | • | • | • |
| Power Budget | | • | • | • | • | • |
| Requirements Verification Matrix (RVM) | | • | • | • | • | • |
| Review Presentation | • | • | • | • | • | • |
| Schedule | • | • | • | • | • | • |
| Software Design Document | | • | • | • | • | • |
| Structural Analysis | | | | • | • | • |
| Avionics Unit, Subsystem, and System-Level Verification Test Procedures and Results | | | • | • | • | • |
| Thermal Analysis | | | • | • | • | • |
| Waivers | | | | • | | |

### 4.8.2 Integration and Test Deliverables Checklist

Table 4-4 indicates when each integration and test deliverable must be submitted to UNP.

Some documents required in earlier stages are no longer required in the integration and test process. All documents should be kept in organized records, and many should continue to be updated. Small design changes remain common in these later development phases, and mission level, system level, and design documents should reflect this. Test results and as-run procedures are also critically important records.

Table 4-4. Integration and Test Deliverables Checklist

| Deliverable | Deep Dive | PIR | PSR |
|---|---|---|---|
| Attitude Determination & Control Subsystem Verification Test Deliverables | • | • | • |
| Assembly Procedures | | • | • |
| Command and Telemetry List | • | • | • |
| Command Execution Test Deliverables | • | • | • |
| Complete Charge Cycle Deliverables | • | • | • |
| Data Budget | • | • | • |
| Day in the Life Test Deliverables | • | • | • |
| Dashboard | • | • | • |
| Full Functional Test Deliverables | | | • |
| Ground Data System | • | • | • |
| GSE Manual | | | • |
| Link Budget | • | • | • |
| Mass Budget | • | • | • |
| Master Equipment List | • | • | • |
| Materials List | • | • | • |
| Mission Design Document | • | • | • |
| Mission Operations Plan | • | • | • |
| Pointing Budget | • | • | • |
| Power Budget | • | • | • |
| Requirements Verification Matrix (RVM) | • | • | • |
| Review Presentation | • | • | • |
| Schedule | • | • | • |
| Simulated Communications Test Deliverables | • | • | • |
| Software Design Document | • | • | • |
| Support Plan | | • | • |
| Avionics Unit, Subsystem, and System-Level Verification Test Procedures and Results | • | • | • |
| Waivers | • | | |

# 5 Satellite Development Guidelines

## 5.1 How do we use these guidelines?

UNP provides the following satellite development guidelines to convey requirements, suggestions, and best practices to teams developing small satellites. Satellites meeting these guidelines will be easier to test, integrate, and fly and see reduced scrutiny in launch and range safety approval. Sources for these guidelines include common Launch Vehicle (LV) and dispenser constraints, recommended standards for safety, and lessons learned from UNP missions. It is expected that all relevant guidelines, denoted by UNP-##, be incorporated into each team's requirements verification matrix. Specifically:

- Any guideline that includes "shall" in the wording shall be copied verbatim into the university's RVM and implemented as a requirement.
- Any guideline that does not include "shall" in the wording will be adapted as appropriate into the university's RVM and/or be implemented, as appropriate, in the university program. The guideline must be followed, but there is some flexibility.

Universities in UNP are the owners of their satellites and are therefore responsible for ensuring all aspects of the spacecraft development, including the delivered hardware and software from third parties, are functional and meet requirements. Ultimately, the university is responsible for ensuring the satellite follows all external (UNP or launch vehicle) and internal (mission specific) requirements and constraints.

## 5.2 Waivers

If the satellite design or programmatic approach is not compliant with any guideline in this chapter, the university is required to submit a waiver request to UNP. UNP will handle waiver requests on a case-by-case basis.

Unless otherwise noted, identifying needed waivers, and submitting waiver requests should occur no later than the Preliminary Design Review (PDR). Additionally, there are several guidelines that identify required discussion topics with UNP. Unless otherwise noted, those discussions should also occur no later than PDR. If noncompliance is found after PDR, waiver requests must still be submitted.

## 5.3 UNP Guidelines

While many of these guidelines will not be applicable to participants in the Mission Concept Program, they are provided for reference.

### 5.3.1 Space Vehicle Guidelines

The purpose of these guidelines is to ensure that the spacecraft will do no harm to the launch vehicle or other spacecraft on the rocket. This is the minimum required of every satellite. The structural subsystem is safety critical and subject to increased scrutiny.

UNP11-1 The CubeSat shall be designed to withstand the launch and on-orbit environments of the launch vehicle without failure that results in damage of the launch vehicle and its contents or failure that causes injury to the ground handling crew.

It is anticipated that all UNP CubeSats will be contained in a dispenser. Dispenser providers set CubeSat requirements on factors such as allowable mass, envelope, center of gravity (CG), satellite stiffness, deployables, Electrical Ground Support Equipment (EGSE), and coordinate reference frame. UNP recommends designing CubeSats to fit the Calpoly CubeSat Design Specification. This includes rail design, protrusions, deployer access points, and mass and center of gravity properties. Conformance with this deployer ensures conformance with most deployers on the market, increasing launch opportunities. Track requirements from your chosen deployer specification in your RVM

UNP11-2 The CubeSat should be designed to meet the requirements specified in the Calpoly CubeSat Design Specification.

### 5.3.2 Structural Guidelines

Since CubeSats must survive launch, the energy input requires consideration. This depends on the launch vehicle, location of the satellite on the launch vehicle, the mounting method, and other factors. Many guidelines provide worst-case inputs with a built-in factor of safety. Factors of safety do not need to be added to the UNP requirements unless specified.

UNP11-3 The CubeSat shall be designed for quasi-static loading of 30g acceleration applied independently in each primary axis.

All accelerations should be applied through the center of mass of the analyzed spacecraft. Thermally induced loading should be included in the analysis.

The Air Force Research Laboratory (AFRL) will conduct random vibration testing on the integrated system to qualify the hardware for flight. The specific tests and levels required are levied by the launch vehicle and only available once manifested.

UNP11-4 The CubeSat should be designed to withstand the launch vehicle vibroacoustic environment of 15 $G_{RMS}$ for one minute in each axis.

The shock environment experienced by satellites is highly dependent on the launch vehicle and configuration. Given the canisterized nature of CubeSats on a launch vehicle and the typical arrangement of these deployers, shock is not anticipated to be a significant design consideration.

All environmental testing will occur in the flight configuration. Once a CubeSat has completed environmental testing and been certified for launch, its configuration will not be broken. This means it will not be taken apart or modified in any way without the written approval of UNP and/or the launch provider.

UNP11-5 The CubeSat volume and internal components shall be freely vented volumes to survive ascent de-pressurization environments (i.e., components with a vent-able volume to area of vent openings ratio of less than 2000 inches).

UNP11-6 Venting analysis shall demonstrate a factor of safety of 2.0.

UNP11-7 Use of pressure vessels or sealed compartments is highly discouraged.

Pressure vessels, and pressurized components, as defined in NASA-STD-5003, as well as sealed containers, are highly discouraged and require a waiver from UNP for inclusion in CubeSats.

In addition to the above discussion on pressurized or sealed components, propulsion guidelines should be discussed on a case-by-case basis with UNP.

UNP11-8 All CubeSats containing propulsion systems should contact UNP to discuss necessary requirements for propulsion systems. This includes but is not limited to inhibits, fluid properties, and post-test check outs.

### 5.3.2.1 General Structural Guidelines

The following guidelines are primarily applicable to structural development.

UNP11-9 Use a machined (milled), all-metallic primary structure. Aluminum 6061-T6 is highly recommended. Use of non-metallic primary-structures is highly discouraged.

UNP11-10 Use of bending as a means of forming metallic structures, is highly discouraged.

UNP11-11 Use of epoxies, adhesives, or tape to join structural components, particularly in the primary structure, in the load path of deployable mechanisms, or in the instances where a failure of that part results in a potential hazard is prohibited.

UNP11-12 Use of "soft goods" such as cables, lines, or plastic parts in the primary structure should be discussed with UNP.

UNP11-13 Use of deployable mechanisms should be discussed with UNP.

UNP11-14 Retaining devices for stowed components that rely solely on friction as a means of retention are prohibited.

UNP11-15 Use of pyrotechnic devices/mechanisms is prohibited. Use of vendor-supplied, non-pyrotechnic mechanisms is allowed with UNP review and waiver (i.e. separation mechanisms for deployable solar panels, etc.).

UNP11-16 Use of welded joints or cast metallic components is prohibited.

UNP11-17 Use of parts or assemblies for which safety is highly dependent upon the build or assembly process is highly discouraged. Specifically, use of composites or rivets is prohibited. Obtain UNP approval for 3D printed parts.

UNP11-18 Use threaded fasteners along with proper torque application and one additional form of back-out protection. Preferred methods of backout protection include Loctite, locking helicoils, and staking the head of socket head cap screws of size #4 and smaller.

UNP11-19 It is encouraged to use multiple fasteners for joining components such that failure of one fastener will not cause a hazardous situation. In general, the use of at least four bolts across any interface is recommended to mitigate this concern.

UNP11-20 All primary structure fasteners should be #4 or larger.

UNP11-21 Bolts used in flight hardware shall be lubricated (with space-rated lubricant) and torqued to values specified in MSFC-STD-486B “Standard, Threaded

Fasteners, Torque Limits For". Note that when using locking fasteners, the measured locking torque (running torque) must be added to the torque value specified in MSFC-STD-486B to calculate the final "wrench torque."

UNP11-22 Ensure all assembly and integration operations are reversible.

UNP11-23 Implement fracture control in accordance with NASA-STD-5003 from the outset of design. Ensure that there are no fracture-critical components in the design and that it is easy to prove that items are non-fracture-critical. Examples are: (1) structures designed with redundant load paths and (2) structures built from machined (milled) metals with well-understood properties and having low stresses.

#### 5.3.2.2 Mechanical Ground Support Equipment (MGSE)

Mechanical Ground Support Equipment (MGSE) is an important factor in safely handling your CubeSat throughout its development process. At minimum, consider the following cases for appropriate CubeSat handling: tabletop work including turning the satellite over, shipping and transportation, test fixtures, integration/de-integration with the dispenser, and lifting to different workspaces. Some canisterized satellites may benefit from an MGSE handle, though this is not required. This handle can be removable or may be integrated into the spacecraft. MGSE that remains attached to the CubeSat for flight must meet all requirements for flight hardware.

UNP11-24 Universities shall provide any required Mechanical Ground Support Equipment (MGSE) for use in assembly, integration and test operations.

UNP11-25 A CubeSat's Mechanical Ground Support Equipment (MGSE) shall be designed such that the satellite can be lifted in any axis.

UNP11-26 Mechanical Ground Support Equipment (MGSE) for 12U CubeSats shall be designed for crane lifting and movement.

UNP11-27 All Mechanical Ground Support Equipment (MGSE) shall be designed using a factor of safety of 5.0 for ultimate failure and be proof loaded to twice the expected satellite load.

### 5.3.3 Material Guidelines

Universities are required to develop and maintain a material list for all flight hardware. It should contain all materials in all satellite components, university developed, or vendor supplied. The list must include not only primary materials, but also coatings such as anodize, plating, iridite, conformal coating, etc. Typically, satellites are exempt from listing conformal-coated electronics and very small mass-produced electronic components such as surface-mount resistors and capacitors. Exceptions include larger, non-hermetically sealed electronic components, which may contain air, or electrical components containing relatively large amounts of adhesives, etc. A materials list must be generated for such components, or the components must be potted/conformal coated. The material list should be regularly reviewed by the university team for compliance with outgassing, corrosion resistance, and flammability resistance requirements. Universities must request a waiver for materials that do not comply with requirements.

Materials with high resistance to stress corrosion cracking should be used wherever possible. Non-metallic materials must comply with outgassing restrictions on maximum collectable volatile condensable material (CVCM) and total mass loss (TML). See the resources section for websites containing material selection, outgassing, and stress corrosion cracking information.

| | |
|---|---|
| UNP11-28 | Use of non-metallic material shall be restricted to materials that have a maximum collectable volatile condensable material (CVCM) content of 0.1% or less and a total mass loss (TML) of 1.0% or less. Use of Loctite 242 and 271 are the only pre-approved exceptions to this requirement. |
| UNP11-29 | Spacecraft materials shall be chosen and constructed such that any component will not reach earth with greater than 15 joules of energy or risk of human casualty above 1:10000 upon atmospheric reentry. |

Use of high-melting point materials which could survive atmospheric reentry, such as titanium and tungsten, presents a safety concern which must be evaluated. Verification of this requirement should be performed with NASA's Debris Assessment Software

| | |
|---|---|
| UNP11-30 | Use of toxic and/or volatile fluids or gasses is prohibited. Use of any material that can undergo a phase change in the launch or on-orbit environment (e.g. water) is discouraged. |
| UNP11-31 | Before purchasing hardware, especially components that contain non-metallic items, obtain a materials list first and ensure that all materials meet the outgassing requirements. |
| UNP11-32 | Where glass must be used, it should be non-pressurized and subject only to inertial loading (ref. *NASA-STD-5003, Section 4.2.3.6*); however, use of glass should be minimized. |

### 5.3.4 Thermal Design Guidelines

CubeSats must remain withing proper temperature ranges during all mission phases. This should be verified with a thermal analysis. Thermal environments are dependent on the integrated launch vehicle configuration, ground operations, and orbital environment. Universities are strongly encouraged to design for all possible low Earth orbit altitudes and inclinations.

Temperature sensors are needed to provide state of health data for sensitive components, such as the battery, on orbit and to support thermal vacuum testing at AFRL. Temperature sensors taking advantage of the one wire bus, which can be used by either the spacecraft (if the CubeSat is on) or the EGSE (if the CubeSat is off), are preferred, so the same temperature sensors can be used.

| | |
|---|---|
| UNP11-33 | Temperature sensors shall be installed on each critical component within the satellite and shall be usable during thermal testing when the satellite is powered on and off. |
| UNP11-34 | If different internal temperature sensors are used for the spacecraft and Electrical Ground Support Equipment (EGSE), the EGSE temperature sensor wires must be terminated at a connector accessible from the exterior of the spacecraft while in the dispenser. |

### 5.3.5 Electrical Design Guidelines

CubeSat designs must address standard electrical/power system safety hazards including shorting, which may lead to fire or ignition sources, routing of wiring, such that failures in one circuit will not affect the safety features in physically adjacent circuits, and battery hazards such as overcharging, shorting, cell reversal, thermally induced failures, and inadvertent activation of hazardous subsystems. Electrical design guidelines are applicable to any circuitry built by the university or supplied by a vendor both in the satellite and the supporting equipment.

#### 5.3.5.1 Wiring and Connector Guidelines

UNP11-35 All connectors on the space vehicle shall be kept within their respective connector mating cycle tolerances.

UNP11-36 All wiring shall be stranded copper with PTFE or ETFE insulation. Temperature sensors are the only pre-approved exemption to the use of copper.

UNP11-37 The space vehicle side of the Electrical Ground Support Equipment (EGSE) interface shall be protected from shorting.

UNP11-38 The space vehicle side of the Electrical Ground Support Equipment (EGSE) interface shall include an "insert-before-flight" closeout.

UNP11-39 Usage of solder cups is discouraged.

UNP11-40 All connectors both connecting to Electrical Ground Support Equipment (EGSE) and within the spacecraft shall have locking mechanisms such that connectors cannot de-mate in flight.

UNP11-41 Electrical Ground Support Equipment (EGSE) connections to the spacecraft shall be keyed.

#### 5.3.5.2 Electrical Bonding and Grounding

UNP11-42 The CubeSat circuit/electrical ground shall physically connect to the spacecraft structure.

UNP11-43 The spacecraft shall have the capability to be grounded during testing via an interface on the structure. At minimum, a #2 threaded bolt hole that remains accessible in the final spacecraft configuration should be included as a grounding strap connection point.

#### 5.3.5.3 Inhibits

Inhibits are a critical safety feature which limit potential hazards to the launch vehicle by preventing restricted satellite functionality until after deployment. These guidelines follow the most restrictive inhibit requirements seen in the launch industry, ensuring launch options remain open. Discuss your inhibit approach with UNP to ensure compliance with these guidelines.

UNP11-44 All inhibit designs should be reviewed and approved by UNP.

UNP11-45 Inhibits shall be independent.

There are three types of electrical inhibits required on the spacecraft: activation, power-rail, and installation inhibits. Activation inhibits, commonly known as separation switches, are the devices that determine the state of the control signals for the power-rail inhibits. Power-rail inhibits break the connection between a source and a load. Installation inhibits are externally accessible controls that ensure the vehicle remains off during ground operations, regardless of activation inhibit state. The electrical inhibits topology shown in Figure 5-1 is recommended, though not the only compliant architecture possible.

UNP11-46 The activation controls for the inhibits shall be two-fault tolerant.

Two-fault tolerance means that two of the inhibits can fail to an uninhibited state and the vehicle remains off. For example, if there are three independent switches wired in series and held in an open state, two may fail to a closed state but the third will keep the vehicle off. This meets two-fault tolerance.

UNP11-47 The power-path inhibits on the solar arrays shall be two-fault tolerant between the panel or Maximum Power Point Tracking (MPPT) output and any load (battery or subsystems).

If there are multiple panels/MPPTs it is acceptable to tie all the outputs together and then inhibit that common node from the load.

UNP11-48 The batteries shall have two high-side inhibits and one low-side inhibit on the power-path between the battery and any load.

A high-side power-path inhibit breaks the connection on the positive side of the circuit (i.e. between the positive terminal of the battery and the bus supply). A low side inhibit disconnects the negative terminal of the battery pack from the bus ground.

UNP11-49 An installation inhibit shall be included in series with the inhibits on the spacecraft, such that the installation inhibit alone is sufficient to keep the spacecraft powered off.

Generally, an installation inhibit is referred to as a “remove before flight” (RBF) or “install before flight” (IBF) feature. This installation inhibit shall not be counted toward the activation or power-rail fault tolerance requirements. The RBF/IBF feature ensures that the satellite will not power on during storage, transportation, or handling regardless of the separation switch state on the satellite. This installation inhibit will be removed or installed before flight, as appropriate, ensuring the activation inhibits may power the satellite upon deployment.

UNP11-50 The Remove Before Flight (RBF)/Install Before Flight (IBF) installation inhibit shall be removable/installable while the CubeSat is in the dispenser.

Positive verification of inhibit state is required. There shall be a capability to detect inhibit state through the EGSE.

UNP11-51 Electrical Ground Support Equipment (EGSE) shall be able to verify the state of the inhibits.

Operator controls are any scenario where an operator must maintain the state of an inhibit, therefore shall not be counted as an inhibit. In addition, software controls such as timers are not counted as inhibits.

UNP11-52 Operator and/or software controls shall not be considered a design inhibit.

Figure 5-1 depicts an example inhibit topology which meets the requirements for any launch vehicle. V_MPPT is the common output of any solar panels or maximum power point trackers (MPPTs), BAT+ and BAT- are the connections to the battery pack, and UNREG is the supply rail to the primary regulators and bus. The symbols show enhancement-mode NMOS transistors which require a gate driver. This implementation is common but not required. The functionality is the following:

- Two-fault tolerance from V_MPPT to the battery is met by A, B, $C_B$, and E.
    - This path is three-fault tolerant, but eliminating an inhibit would break other functions.
    - $C_A$ does not contribute to fault tolerance in this case; the body diode will conduct.
    - Fault tolerance on the diode-OR is met by D2, D3, and E.
- Two-fault tolerance from V_MPPT to the system is met by A, B, and D.
    - F does not contribute to fault tolerance as the control signal (RBF) is asserted in flight configuration.
- Two-fault tolerance from the battery to the system met by $C_A$ , D, and E.
    - $C_B$ does not contribute to fault tolerance in this case; the body diode will conduct.
    - F does not contribute to fault tolerance as the control signal (RBF) is asserted in flight configuration.

One acceptable variation is to control the MOSFETs via independent switch signals rather than a common enable signal. Another is to use the RBF to control the enable signal instead of a dedicated MOSFET.

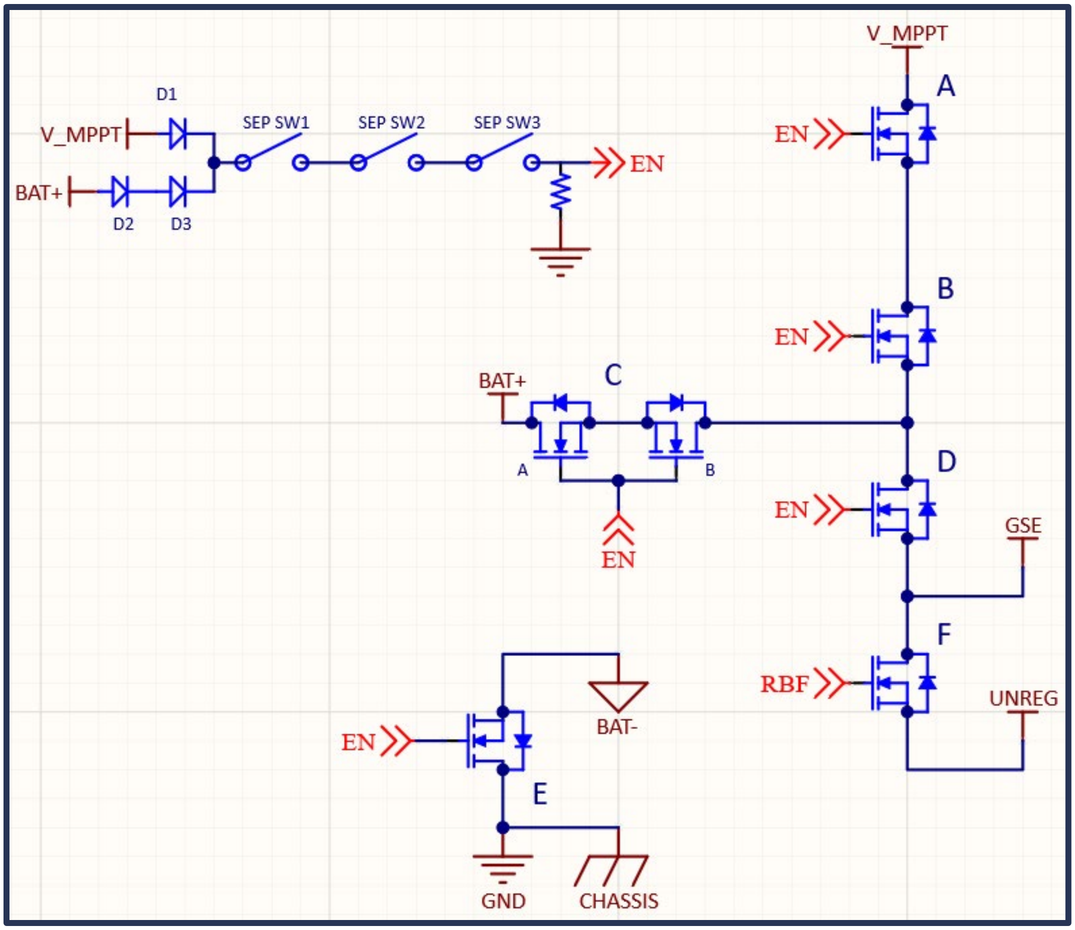


Figure 5-1 Inhibit Scheme Example

### 5.3.5.4 Battery Design

Many universities will choose COTS batteries designed for CubeSats. If building your own battery pack, careful consideration should be paid to the choice of cells. There are several flight-qualified batteries of various chemistries procurable by universities. Universities are responsible for battery and electrical system safety. Whether in-house or COTS batteries, keeping the pack below 80 watt-hours reduces launch safety concerns.

UNP11-53 Batteries should be selected from the Underwriters Laboratory (UL) approved list.

UNP11-54 Battery capacity should be less than or equal to 80 W-hr.

### 5.3.5.5 Electromagnetic Interference and Electromagnetic Compatibility (EMI/EMC)

The CubeSat design should consider electromagnetic compatibility (EMC) and mitigation of electromagnetic interference (EMI), specifically susceptibility to launch vehicle and range radiation environments. Universities are not required to generate a detailed EMI/EMC analysis but should be aware of potential EMI/EMC concerns. EMI/EMC concerns are particularly applicable to CubeSats due to the proximity of components.

UNP11-55 CubeSat design should consider Electromagnetic Compatibility (EMC), Electromagnetic Interference (EMI) mitigation, and radiation.

### 5.3.5.6 General Hardware Practices

UNP11-56 CubeSat designs should incorporate basic health monitoring of all hardware. Health statistics generally include temperatures, currents, and voltages. Current and voltage measurements should be tracked for independent solar panels, the battery, power distribution busses, and individual components.

### 5.3.5.7 Electrical Ground Support Equipment (EGSE)

#### 5.3.5.7.1 EGSE Functional Requirements

UNP11-57 Electrical Ground Support Equipment (EGSE) shall be capable of performing battery charging and discharging with the satellite inhibited.

UNP11-58 Electrical Ground Support Equipment (EGSE) shall be capable of performing inhibit actuation (set/reset inhibits).

UNP11-59 Electrical Ground Support Equipment (EGSE) shall be capable of powering the satellite while the satellite is inhibited.

UNP11-60 Electrical Ground Support Equipment (EGSE) shall be capable of supporting functional testing of the satellite, including subsystem level, full functional testing, and full “day in the life” testing.

The EGSE will follow the spacecraft through final testing and integration. Portable, robust designs, such as those built into “military type” boxes, like pelican cases, are recommended.

UNP11-61 Electrical Ground Support Equipment (EGSE) shall be self-contained and portable.

UNP11-62 Electrical Ground Support Equipment (EGSE) shall be capable of satellite thermal monitoring.

#### 5.3.5.7.2 EGSE Communications

EGSE shall be capable of command and control of the satellite without the radio. EGSE shall also be capable of command and control of the satellite through radios and Radio Frequency (RF). Both communications channels must be available due to various integration and test environments. It must be possible to perform integration and test without free radiation as this is not allowed in many facilities. Functional testing must exercise all hardware, including the RF communications subsystem. Antenna hats are recommended for this purpose. Functional testing after integration to the launch vehicle, if performed, must not generate free radiation of RF energy.

UNP11-63 Electrical Ground Support Equipment (EGSE) shall be capable of command and control of the satellite without the radio.

UNP11-64 Electrical Ground Support Equipment (EGSE) shall be capable of command and control of the satellite through radios and Radio Frequency (RF).

#### 5.3.5.7.3 EGSE Satellite Battery Charging Capability

UNP11-65 Satellite battery charging/discharging equipment in the Electrical Ground Support Equipment (EGSE) shall be current limited by design.

UNP11-66 Electrical Ground Support Equipment (EGSE) shall be capable of charging and discharging the satellite battery without enabling the satellite bus/loads.

#### 5.3.5.7.4 EGSE Switches and Controls

UNP11-67 The main power switch shall be provided with an indicator light.

UNP11-68 All Electrical Ground Support Equipment (EGSE) switches or buttons shall be clearly labeled.

UNP11-69 Separation between switches/buttons shall be sufficient to avoid accidental actuation.

UNP11-70 Switches should include covers, with an automatic power-off feature, such that when the cover is closed the switch is in the off position.

#### 5.3.5.7.5 Powering EGSE

UNP11-71 Electrical Ground Support Equipment (EGSE) shall use standard 120 V, 60 Hz, 3 prong “household” power, preferably through a single plug.

#### 5.3.5.7.6 EGSE Safety Requirements

UNP11-72 All Electrical Ground Support Equipment (EGSE) connectors that interface with flight hardware shall be designed, built, and controlled in a manner commensurate with flight hardware except for cleanliness and, therefore, must meet applicable requirements for configuration control, quality assurance, parts and materials, etc.

#### 5.3.5.7.7 EGSE Circuit Protection

UNP11-73 Circuit protection, including over voltage, reverse voltage, and over current protection shall be installed on all primary circuits/load lines on the satellite and on all external ground support equipment interfacing with the satellite.

UNP11-74 Circuit protection devices shall be readily accessible for inspection, reset, or replacement.

UNP11-75 Circuit protection shall be clearly marked with rated voltage and amperage.

### 5.3.6 Communication Guidelines

The satellite must be designed with a robust communications link for command and control.

UNP11-76 There shall be a minimum of 6dB margin in the telecommunications link analysis both for the uplink and the downlink at a 10-degree elevation mask.

One of the requirements from the Federal Communications Commission (FCC) is the ability to cease transmission if it is determined that a satellite is causing harmful interference.

UNP11-77 Satellites shall be capable of ceasing transmission if required to do so by the government.

#### 5.3.6.1 Encryption

Encryption comes in many types, depending on the satellite's capabilities and mission, and is required for satellites participating in UNP. Encryption required on uplink to ensure satellite control is secure. For satellites performing proximity operations or taking images, encryption is required on uplink and downlink. Technical demonstrations may require encryption on the downlink and will be dealt with on a case-by-case basis by UNP.

UNP11-78 Uplink communications shall be encrypted for all satellites.

UNP11-79 Downlink encryption approach should be presented to and approved by UNP.

The Advanced Encryption Standard (AES) algorithm is recommended by UNP. There are various operating modes of the AES algorithm, such as Electronic Code Book (ECB), Cypher Block Chaining (CBC), and Counter (CTR). Which mode and length the university should choose depends on the mission, but in general AES-128 Galois/Counter Mode (GCM) with a 128-bit key is recommended.

Please note that neither UNP nor AFRL can directly assist in FCC licensing or International Amateur Radio Union (IARU) coordination on behalf of any UNP team.

### 5.3.7 Handling Requirements

#### 5.3.7.1 Cleanliness Requirements

Cleanliness is an important consideration throughout flight assembly, integration, test, and transport to AFRL and the launch site. While UNP provides a minimum requirement, certain missions may require higher levels of cleanliness.

UNP11-80 The flight hardware shall be maintained in a class 100,000 level or better facility as defined in FED-STD-209E.

UNP11-81 All flight hardware shall be maintained at the Visibly Clean (VC) level (free from manufacturing residue, dirt, oil, grease, processing debris, or other visible particulate when inspected with the unaided or corrected-vision eye).

### 5.3.8 On-Orbit Operational Requirements

On-orbit requirements are launch vehicle dependent. Additional requirements may be enforced once a satellite is manifested. Requirements in this section have been generalized to encompass most launch vehicles. Universities will be held to these requirements during design; however, the specific requirements are subject to change.

UNP11-82 The CubeSat should be designed for variable delays after separation from the launch vehicle of the following: start of Radio Frequency (RF) transmission, active deployable actuation, attitude maneuvers, and orbital maneuvers. Final delay times should be coordinated with UNP.

#### 5.3.8.1 Orbital Requirements

It is advantageous to be able to achieve full mission success in a variety of orbital altitudes and inclinations. Teams should perform analyses to understand the impact of various orbits on full and minimum mission success criteria. Orbital flexibility greatly widens launch opportunities.

Current US national policy requires deorbit/disposal within 5 years from end-of-life. If a high altitude/long lifetime orbit is required for mission success, then a means of decreasing orbital lifetime must be incorporated into the satellite.

UNP11-83 CubeSat teams should perform a detailed orbital analysis to understand the impacts of altitude / inclination on mission success; and coordinate with UNP to understand evolving US national policy deorbit requirements.

#### 5.3.8.2 Ground Control

Universities are responsible for ground control of the satellite after deployment from the launch vehicle. Universities are also responsible for obtaining the necessary spectrum licenses for operating their ground and space radio communications equipment. Universities should not try to obtain spectrum licenses until after FSR, but understanding the license process will be important during hardware trade studies. The university has satellite control authority (SCA). Please note that neither UNP nor AFRL can directly assist in FCC licensing or IARU coordination on behalf of any UNP team.

#### 5.3.8.3 End-of-Life Operations

CubeSats should incorporate means to mitigate common causes of on-orbit debris generating events and hazardous occurrences. The vehicle must be made safe at end-of-life. This includes turning off radio transmissions, venting propellant, and deorbiting/moving to a disposal orbit.

UNP11-84 CubeSats should incorporate means to mitigate common causes of on-orbit debris generating events or hazardous occurrences.