

University of Colorado
Department of Aerospace Engineering Sciences
Senior Projects - ASEN 4018

Lockheed Martin's satellite Light Adcs fault Management System (LLAMAS)
Conceptual Design Document

Monday 2nd October, 2017

1. Information

1.1. Project Customers

| |
|---|
| Name: Dr. Jim Chapel Email: jim.d.chapel@lmco.com Phone: 303-977-9462 |
|---|

1.2. Team LLAMAS Members

| | |
|---|---|
| Name: Dalton Anderson Email: dalton.anderson@colorado.edu Phone: 720-438-5449 | Name: Sam O'Donnell Email: samuel.odonnell@colorado.edu Phone: 858-699-5201 |
| Name: Dan Greer Email: daniel.greer-1@colorado.edu Phone: 720-320-6505 | Name: Zach Reynolds Email: zare0851@colorado.edu Phone: 303-263-0231 |
| Name: Ben Hutchinson Email: behu1036@colorado.edu Phone: 308-250-0077 | Name: Kristyn Sample Email: kristyn.sample@colorado.edu Phone: 573-819-9956 |
| Name: Kent Lee Email: Kent.lee@colorado.edu Phone: 502-523-6219 | Name: Corwin Sheahan Email: corwin.sheahan@colorado.edu Phone: 661-912-2204 |
| Name: Andrew Levandoski Email: Andrew.Levandoski@colorado.edu Phone: 303-885-8519 | Name: Pol Sieira Email: Pol.Sieira@colorado.edu Phone: 720-210-3314 |
| Name: Andrew Mezich Email: Andrew.Mezich@colorado.edu Phone: 907-903-4141 | Name: Zack Toelkes Email: zato0770@colorado.edu Phone: 785-393-3762 |

Contents

| | | |
|----------|---|-----------|
| 1 | Information | 1 |
| 1.1 | Project Customers | 1 |
| 1.2 | Team LLAMAS Members | 1 |
| 2 | Project Description | 4 |
| 2.1 | Purpose and Objective | 4 |
| 2.2 | CONOPS | 6 |
| 2.3 | Functional Requirements | 7 |
| 2.4 | Functional Block Diagram | 7 |
| 3 | Project Requirements | 7 |
| 4 | Key Design Options Considered | 12 |
| 4.1 | TestTable | 12 |
| 4.1.1 | Air Table | 12 |
| 4.1.2 | Ice Table | 13 |
| 4.1.3 | Air Bearings | 13 |
| 4.2 | Bounded Translation | 14 |
| 4.2.1 | Bounded Translation | 15 |
| 4.2.2 | Diagrams and Description of Bounded Area | 15 |
| 4.3 | Attitude Determination Sensors | 17 |
| 4.3.1 | Relative Sensors | 17 |
| 4.3.2 | Absolute Sensors | 18 |
| 4.4 | MockSat Processor | 19 |
| 4.5 | Fault Injection System | 20 |
| 4.5.1 | Fault Actuation System | 20 |
| 4.5.2 | Fault Injection Software on Single MCU | 21 |
| 4.5.3 | Fault Injection Software on Separate MCU Option 1 | 22 |
| 4.5.4 | Fault Injection Software on Separate MCU Option 2 | 22 |
| 4.5.5 | Pros and Cons Table | 23 |
| 4.6 | Fault Management Software | 23 |
| 4.7 | Battery Types | 24 |
| 4.7.1 | Lithium Based Rechargeable Batteries | 24 |
| 4.7.2 | Nickel Based Rechargeable Batteries | 25 |
| 4.7.3 | Non-rechargeable Batteries | 25 |
| 4.7.4 | Pros and Cons Table | 25 |
| 4.7.5 | Power Flow | 25 |
| 4.8 | Communications | 26 |
| 5 | Trade Study Process and Results | 27 |
| 5.1 | TestTable | 27 |
| 5.2 | Bounded Translation | 28 |
| 5.3 | Attitude Determination Sensors | 29 |
| 5.4 | Fault Injection System | 30 |
| 5.5 | Battery Types | 30 |
| 6 | Selection of Baseline Design | 31 |
| 6.1 | Overall System Design | 31 |
| 6.2 | TestTable | 31 |
| 6.3 | Bounded Translation | 32 |
| 6.4 | Attitude Determination Sensors | 32 |
| 6.4.1 | Relative Sensors | 32 |
| 6.4.2 | Coarse Sensor | 32 |
| 6.4.3 | Fine Sensor | 32 |
| 6.5 | Battery Type | 32 |

| | | |
|------------|---|----|
| 6.6 | Fault Injection and Fault Management Software | 32 |
| 6.7 | MockSat Processor | 33 |
| 6.8 | Communications | 33 |
| References | | 34 |

2. Project Description

2.1. Purpose and Objective

The motivation of this project is grounded in the identification and mitigation of operational faults, more specifically, those which occur in the avionics of satellites. As a response to the likelihood of satellites to fall victim to faulty avionics systems, the ability to detect, analyze, and respond to faults in a way that maintains the satellite's operational integrity is becoming an increasingly desirable technology. Common practice is to specifically tailor the process of testing a satellite's avionics for fault responses to the individual satellite. The ability to modulate this process in a way that it can be applied to the avionics of satellites across the board reduces the cost and man hours involved in testing.

The overarching aim of this project is to design a system that meets the goals outlined by Lockheed Martin. The system will consist of 2 primary elements: MockSat and TestTable.

The MockSat will model the rotational dynamics of GOES-16 satellite in restricted planar motion. The MockSat will be scaled to allow for operation on the TestTable. It will be equipped with an attitude determination and control system (ADCS) which uses hardware similar to that of the GOES-16 satellite. The MockSat sensor array will include a coarse attitude sensor, a fine attitude sensor, and a relative attitude sensor. These three sensors will work in conjunction to determine the satellite's attitude relative to an arbitrary reference point. The MockSat will also have redundant reaction wheels to achieve control its angular dynamics. The reaction wheels will be chosen to allow the bandwidth response of the MockSat to approximately replicate the 0.04 Hz bandwidth response of the GOES-16 satellite. Redundant reaction wheels will allow for injection of a predefined fault in order to evaluate the fault response of the flight software. The resulting flight data of the MockSat will be retained for future analysis¹. Future projects may mount translational actuators (i.e. cold gas thrusters) to the MockSat to control its translational dynamics, This system will restrict translation via a removable external apparatus. This imposed constraint on translational motion is justified since the primary purpose of the translational actuators on the GOES-16 is station keeping, which maps to holding the MockSat stationary. In addition to the ADCS, the MockSat will have onboard processing, data storage, communications, and power regulation capabilities. Potential options for all of these subsystems are discussed in Sections 4-6.

The TestTable will be designed such that it supports the operations of the MockSat by providing an environment that places limited restrictions on the planar dynamics and motion of the MockSat when placed on the TestTable surface. This limited restriction environment will be referred to as "frictionless" throughout this document. While the MockSat will only have onboard attitude control, the TestBed will allow for support of a future MockSat design that has translation control as well as attitude control.

The ADCS in modern space vehicles are designed to be single fault tolerant, which is characterized by the following:

1. Zero single points of failure (SPOF). No single failure in any one component should be so catastrophic as to disable the system.
2. Identification of the failed component. The fault should be classified to the greatest extent possible in order to optimize the fault response strategy.
3. Isolation/containment of the failure. In order to protect against fault propagation, the faulty component should be isolated in the most comprehensive manner possible.
4. The ability to revert to a safe mode. No failure should initiate a cascade failure that results in the system being disabled.

Since single fault tolerant flight software is complex and difficult to test, a simple, modular system is desirable for rapid testing of flight software configurations. By separating the ADCS software from fault management software, fault testing becomes increasingly modular, allowing for simpler and less costly design.

Table 1 breaks down this project's success into level 1, 2, and 3 objectives; level 1 indicates the minimum definition of success, and level 3 is the primary success metric for each major subsystem. The TestTable will allow three degrees of freedom (3DOF) for the MockSat (two translational and one rotational). Translation of the MockSat will be bounded in order to provide station keeping (keeping the MockSat centered in the testing area), whereas GOES-16 uses thrusters for this purpose.

Table 1. Level 1, 2, and 3 Objectives

| | Level 1 | Level 2 | Level 3 |
|--------------------|--|---|---|
| TestTable | Construct a TestTable to allow for 2D translation dynamics with passive control, 1D rotation dynamics, support weight of MockSat, stationary attitude reference | | Moving attitude reference |
| MockSat Hardware | Power source, position sensor, coarse orientation sensor (analogous to sun sensor), fine orientation sensor (analogous to star tracker), redundant reaction wheels (1 set of 2), ADCS/fault injection processor, data storage, 15 minute constant operating time | 30 minute constant operating time | 60 minute constant operating time |
| Fault Injection | Inject fatal operating fault into primary reaction wheel after a pre-determined time from testing start | Inject fatal operating fault into coarse orientation sensor | Inject fatal operating fault into fine orientation sensor |
| Fault Management | Upon fault injection, the MockSat will recognize the presence of the fault and enter a safe mode | | Upon user command, MockSat responds to injected fault in a way that maintains operational integrity |
| MockSat Control | Active planar rotational control with passive translational control | | |
| Comm/Data Handling | Flight software uploaded prior to testing; fault uploaded prior to testing; telemetry data stored onboard MockSat; Ground Station data analysis post-test | Wired, real-time telemetry and fault injection | Wireless, real-time telemetry and fault injection |

The MockSat will have redundant reaction wheels (primary and secondary) for attitude actuation, at a minimum. The reaction wheels will provide an attitude control bandwidth response of $\pm 10\%$ of 0.04 Hz, which mimics the pointing response of the GOES-16 satellite. For attitude determination, there will be two sensors: one coarse and one fine. These are analogs to sun sensors and star trackers, respectively. A coarse sensor will have a wider angle of view, but less resolution of attitude determination, and the fine sensor will have a more narrow field of view and will provide a higher degree of accuracy. The coarse/fine sensor array will allow the system to mimic the pointing precision of a star tracker, without the need to include a catalog of stars. As part of a successful demonstration, the MockSat will undergo at least two attitude verification tests before fault injection, then will go through at least two attitude verification tests after fault recovery. The attitude determination tests have not been designed specifically, however, they are expected to be on the order of 15 minutes to an hour. As such, it is desired that the MockSat be able to run for up to 60 minutes autonomously as a Level 3 objective. The MockSat will also need an onboard power source to power the flight control and fault injection systems for this time frame.

The MockSat will also have an onboard processor for flight software and fault injection control. The MockSat control software will autonomously maintain a specified orientation to within $\pm 2.5^\circ$ of a desired pointing angle, and be able to autonomously adjust orientation with an attitude control bandwidth response of $\pm 10\%$ of 0.04 Hz. The attitude control system will be single-fault tolerant, meaning that the MockSat flight software will be able to respond to a fatal operating fault in a reaction wheel, the coarse attitude sensor, or the fine attitude sensor (but not more than one fault happening concurrently).

Lastly, the MockSat will have a separate control system onboard to create faults. This system will be able to cause a fatal fault in the primary reaction wheel on the MockSat, as well as fault the coarse and fine attitude sensors. This system will be independent from the MockSat flight control system, and will not interfere with the attitude and location control of the MockSat (other than by causing fatal operating faults). As a bare minimum, the fault injection system will cause a fatal operating fault in the primary reaction wheel after a pre-programmed amount of time. This fault

needs only to prevent the proper operation of the reaction wheel. Therefore, in order to minimize component cost, it is desired to have the fault not permanently damage the primary reaction wheel and ADCS system. As a Level 2 objective, the fault will be able to be triggered by a user on demand, rather than after a predetermined time frame. Along with triggering the fault on demand, it will be able to inject a fault into either the reaction wheel or coarse orientation sensor. Finally, as a Level 3 objective, the fault injection system will have the capability of triggering fatal faults in the fine orientation sensor.

2.2. CONOPS

The concept of operations, shown in Figure 1, begins with upload of flight software to the MockSat to maintain a reference orientation on the low-friction TestTable. Reference target is monitored with fine and coarse sensor, while actuation of the satellite is controlled by reaction wheels. The reference target has the capability of moving to ensure (visually by observers) that the MockSat has proper attitude control, and can track and maintain attitude reference with respect to the target. Normal operation continues, while telemetry data is uplinked to the GSU, until fault is injected from ground station. The MockSat properly identifies and responds to the fault by entering a safe-mode and alerting user of fault, allowing the ground station operator to command the satellite to recover and reorient itself.

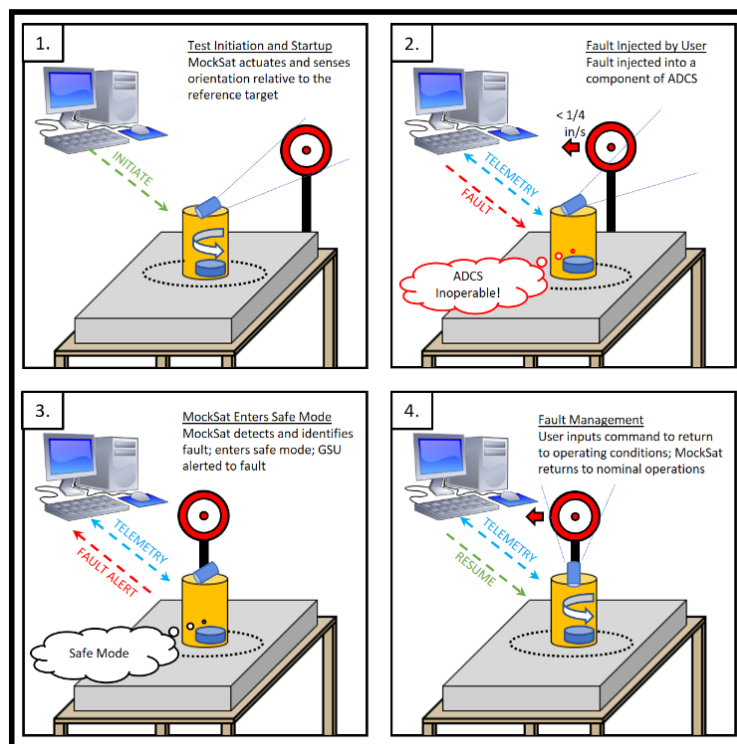
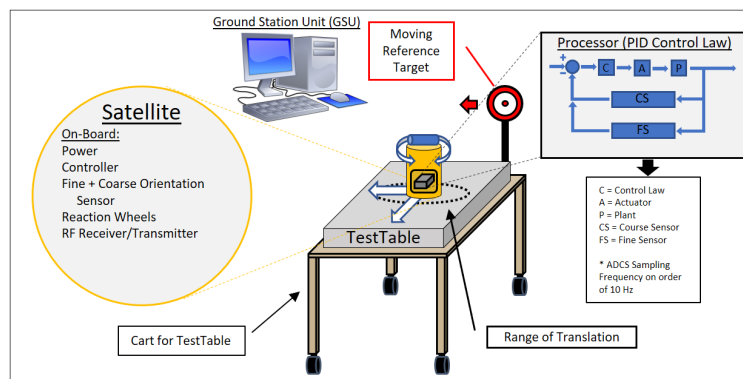


Figure 1. Concept of Operations

2.3. Functional Requirements

1. The TestTable shall allow for two degrees of freedom in translation and one degree of freedom in rotation.
2. The MockSat shall be equipped with an attitude determination and control system (ADCS) that replicates the 0.04 Hz bandwidth response of the GOES-16 satellite to within $\pm 10\%$.
3. The MockSat shall have the ability to maintain a controlled attitude relative to a point of reference within $\pm 2.5^\circ$.
4. The system shall have the ability to introduce a fatal operating fault in either the MockSat's primary reaction wheel, the coarse orientation sensor, or the fine orientation sensor (but not more than one fault at a time).
5. The MockSat flight control software shall recover from a fatal operating fault in either the MockSat's primary reaction wheel, the coarse orientation sensor, or the fine orientation sensor (but not more than one fault at a time) by entering a safe mode.

2.4. Functional Block Diagram

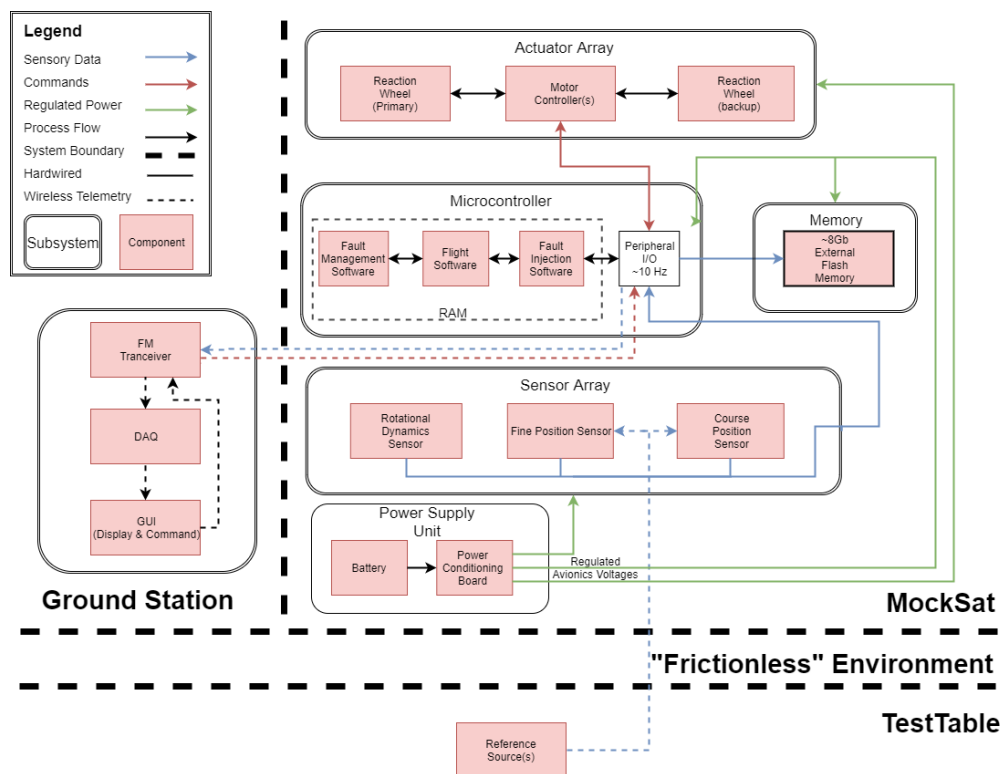


Figure 2. Functional Block Diagram

The functional block diagram provided above identifies how the subsystems and their respective components interact. The primary systems making up the LLAMAS architecture are a ground station, the MockSat, the TestTable, and a frictionless environment. The ground station will allow for wireless communication with the satellite and real-time data display to the user during testing. The MockSat will implement the pre-defined control laws based on the sensory data obtained using the point reference. The TestTable will serve to provide the frictionless environment required by the MockSat. Finally, the frictionless environment will allow for the 3DOF required for testing.

3. Project Requirements

1. The TestTable shall allow for two degrees of freedom in translation and one degree of freedom in rotation.
 - 1.1. The TestTable shall provide for isolated 3DOF dynamics.

- **Motivation:** *To simulate the dynamics of the MockSat in a zero-g environment.*
 - **V&V:** *Inspection.*
- 1.1.1. The TestTable shall allow for simultaneous, unrestricted translation along two orthogonal axes within a designated portion of the plane of the TestTable surface.
- **Motivation:** *To simplify the dynamics of the MockSat to two degrees of translational freedom.*
 - **V&V:** *Inspection. MockSat only moves along surface of TestTable within the designated boundary.*
- 1.1.2. The TestTable shall allow for unrestricted rotation of the MockSat about its axis normal to the plane of the TestTable surface.
- **Motivation:** *To simplify the dynamics of the MockSat to a single rotational degree of freedom.*
 - **V&V:** *Inspection. MockSat only rotates about axis normal to the surface of the TestTable.*
- 1.2. The TestTable shall support the weight of the MockSat whilst providing for a reduced-friction surface.
- **Motivation:** *TestTable surface should not inhibit the motion of the MockSat due to friction.*
 - **V&V:** *Test.*
- 1.2.1. The maximum frictional force between the MockSat and the TestTable during operation shall be no greater than 1% of the maximum frictional force between a block of a representative MockSat material (i.e. aluminum) of a similar mass as the MockSat and a plate of a representative TestTable material (i.e. polycarbonate).
- **Motivation:** *Derived. Required for meeting the criteria of a reduced-friction surface.*
 - **V&V:** *Test/Analysis*
- 1.3. The TestTable shall comply with OSHA Two-Man Lift criteria.
- **Motivation:** *To ensure safe transportation functionality of the TestTable.*
 - **V&V:** *Inspection/Demonstration.*
- 1.3.1. The TestTable shall occupy a volume no greater than 72 x 72 x 28 inches.
- **Motivation:** *Derived. TestTable has to be transportable through standard doorways.*
 - **V&V:** *Inspection.*
- 1.3.2. The TestTable shall weigh no more than 100 pounds.
- **Motivation:** *Derived. To satisfy Requirement 1.3.*
 - **V&V:** *Inspection.*
- 1.4. The TestTable shall have a right-handed inertial reference frame **{X,Y,Z}**.
- **Motivation:** *To provide a consistent reference frame for position and attitude determination.*
 - **V&V:** *Inspection/Analysis.*
- 1.4.1. The orthogonal axes of translation **X** and **Y** shall be oriented such that they are parallel with the sides of the TestTable.
- **Motivation:** *Derived. Required for defining right-handed inertial reference frame.*
 - **V&V:** *Inspection/Analysis.*
- 1.4.2. The origin of the inertial reference frame shall be positioned in the geometric center of the TestTable surface.
- **Motivation:** *Derived. Required for defining right-handed inertial reference frame.*
 - **V&V:** *Inspection/Analysis.*

- 1.4.3. Positive rotation of the MockSat shall be defined as counterclockwise rotation with the rotation vector directed up orthogonal from the TestTable surface.
 - **Motivation:** *Derived. Required for defining right-handed inertial reference frame.*
 - **V&V:** *Inspection/Analysis.*
2. The MockSat shall be equipped with an attitude determination and control system (ADCS) that replicates the 0.04 Hz bandwidth response of the GOES-16 satellite to within $\pm 10\%$.
 - 2.1. The MockSat shall be equipped with two reaction wheels for rotational control.
 - **Motivation:** *GOES-16 uses reaction wheels to actuate, and implements redundant components for mission longevity.*
 - **V&V:** *Inspection*
 - 2.1.1. The MockSat reaction wheels shall be scaled/tuned to simulate the response of GOES-16 about its max MOI.
 - **Motivation:** *Customer specified requirement.*
 - **V&V:** *Analysis. Research will be performed to determine GOES-16 reaction, and calculations will be performed on the MockSat to ensure that the desired response is achieved.*
 - 2.1.2. The MockSat reaction wheels shall be capable of responding to user fault injection.
 - **Motivation:** *Levels of success dictate demonstration of reaction wheel injected failure and response.*
 - **V&V:** *Inspection*
 - 2.2. MockSat shall have a sensor to provide rotational data.
3. The MockSat shall have the ability to maintain a controlled attitude relative to a point of reference within $\pm 2.5^\circ$.
 - 3.1. The MockSat shall be equipped with a sensor array to determine its orientation.
 - **Motivation:** *The GOES-16 uses primarily star trackers to determine its attitude. The MockSat is restricted to planer motion, and thus requires less sophisticated trackers. Additionally, the project does not require the MockSat tracker to mimic the GOES-16 attitude determination system.*
 - **V&V:** *Inspection*
 - 3.1.1. The MockSat shall have a coarse sensor to provide a wide field of view and get fine sensor in range.
 - **Motivation:** *Coarse sensor will provide a field of view to bring fine sensor within visible range.*
 - **V&V:** *Inspection/Analysis*
 - 3.1.2. The MockSat shall have a fine sensor to determine attitude with an accuracy of $\pm 2.5^\circ$.
 - **Motivation:** *Functional requirements dictate a pointing accuracy of $\pm 2.5^\circ$.*
 - **V&V:** *Inspection/Analysis*
 - 3.1.3. The MockSat shall maintain pointing accuracy for no less than 30 seconds.
 - **Motivation:** *To demonstrate that the MockSat has attitude determination and control.*
 - **V&V:** *Analysis. Telemetry data will provide attitude and will allow for deriving pointing accuracy.*
4. The system shall have the ability to introduce a fatal operating fault in either the MockSat's primary reaction wheel, the coarse orientation sensor, or the fine orientation sensor (but not more than one fault at a time).
 - 4.1. The fault injection system shall not cause permanent damage to the ADCS system.
 - **Motivation:** *Want to be able to continue to use the fault management test bed without replacing components regularly.*

- **V&V:** *Test. Verify MockSat can still run nominally after test conclusion and components not visually damaged.*
- 4.2. The fault injection system shall wait for user command from the ground station to initiate fault injection.
- **Motivation:** *Customer specified requirement.*
 - **V&V:** *Demonstration. System will function nominally until fault is injected via telemetry triggered by user.*
- 4.2.1. The ground station unit shall allow the user to initiate a choice of reaction wheel fault, coarse sensor fault, or fine sensor fault.
- **Motivation:** *Derived. Necessary to meet Requirement 4.2, specific faults must be chosen to demonstrate functionality with separate faults.*
 - **V&V:** *Demonstration. For a given fault an specific response is expected and can determine whether the correct fault has been inject.*
- 4.2.2. The ground station unit shall be able to send a command for fault initiation to the fault injection system.
- **Motivation:** *Derived. Necessary to meet Requirement 4.2, communication between the ground station and MockSat is necessary for user command to initiate fault injection.*
 - **V&V:** *Test. Prove MockSat receives signals sent by ground station with simple communication test.*
- 4.3. The fault injection system shall be able to be deactivated by user command.
- **Motivation:** *To be able to stop a fault management demonstration and inject another fault for a new demonstration.*
 - **V&V:** *Test. Use telemetry data to verify that fault disappears once deactivated.*
- 4.3.1. The ground station unit shall allow the user to deactivate the fault injection system
- **Motivation:** *Derived. Necessary for Requirement 4.3, interface on ground station must allow for deactivation of fault injection.*
 - **V&V:** *Test. Stop fault injection from ground station and check data to prove fault injection was properly deactivated.*
- 4.3.2. The ground station unit shall be able to send a command to deactivate the fault injection system.
- **Motivation:** *Derived. Necessary for Requirement 4.3, communication between the ground station and MockSat is necessary for user command to deactivate fault injection.*
 - **V&V:** *Test. Test for Requirement 4.3.1. is sufficient.*
5. The MockSat flight control software shall recover from a fatal operating fault in either the MockSat's primary reaction wheel, the coarse orientation sensor, or the fine orientation sensor (but not more than one fault at a time) by entering a safe mode.
- 5.1. There shall exist in software a fault management system to handle fault detection and identification.
- **Motivation:** *Customer Specified Requirement.*
 - **V&V:** *Inspection. Software will live on MCU.*
- 5.1.1. The fault management system shall have the ability to detect a fatal operating fault from the reaction wheel.
- **Motivation:** *Derived. Customer Specified Requirement. Ultimately the system is able to detect a fault in the reaction wheel and both sensors.*
 - **V&V:** *Test. Verify a fault in the reaction wheel using data and fault classification alert match.*

- 5.1.2. The fault management system shall have the ability to detect a fatal operating fault from the coarse attitude sensor.
- **Motivation:** *Derived. Customer Specified Requirement. Ultimately the system is able to detect a fault in the reaction wheel and both sensors.*
 - **V&V:** *Test. Verify a fault in the reaction wheel using data and fault classification alert match.*
- 5.1.3. The fault management system shall have the ability to detect a fatal operating fault from the fine attitude sensor.
- **Motivation:** *Derived. Customer Specified Requirement. Ultimately the system is able to detect a fault in the reaction wheel and both sensors.*
 - **V&V:** *Test. Verify a fault in the reaction wheel using data and fault classification alert match.*
- 5.1.4. The fault management system shall be agnostic to the fault injection system.
- **Motivation:** *To properly demonstrate how a fault in a satellite will be corrected without complete understanding of how the fault occurred.*
 - **V&V:** *Inspection. Verify by looking at software.*
- 5.1.5. The fault management system shall revert the satellite to a safe mode upon fault detection.
- **Motivation:** *To satisfy the objective of maintaining the satellite in the event of a fault, the satellite must enter a safe-mode.*
 - **V&V:** *Inspection. Inspect that the reaction wheel has turned off.*
- 5.1.5.1. The fault management system shall be able to communicate with the power regulation board.
- **Motivation:** *To allow the fault management system to cut power from primary and secondary reaction wheel or power cycle the reaction wheels and sensors.*
 - **V&V:** *Test. Show that the fault management software can cut power to the primary reaction wheel, the secondary reaction wheel, the coarse sensor, or the fine sensor.*
- 5.1.5.2. The fault management system shall be able to cut off power to the primary reaction wheel.
- **Motivation:** *Derived. Given by requirement 5.1.5.1.*
 - **V&V:** *Test. Show that the fault management software can cut power to the primary reaction wheel.*
- 5.1.5.3. The fault management system shall be able to cut off power to the secondary reaction wheel.
- **Motivation:** *Derived. Given by requirement 5.1.5.1.*
 - **V&V:** *Test. Show that the fault management software can cut power to the secondary reaction wheel.*
- 5.1.6. The fault management system shall alert the ground station operator that a fatal fault has occurred.
- **Motivation:** *The ground station operator must be alerted to a fault before being able to correct the fault from ground station by switching to a redundant reaction wheel or power cycling.*
 - **V&V:** *Test. Test if the fault management system will communicate back to ground station after fault has been injected.*
- 5.1.6.1. The fault management system shall be able to alert the ground station operator to the type of fault that has occurred.
- **Motivation:** *An alert is useless to ground station unless the type of fault can be identified and properly dealt with.*
 - **V&V:** *Test. Compare data from flight to fault indicated by ground station.*

5.1.6.2. The fault management system shall be able to classify the identified fault into one of either the reaction wheel, coarse orientation sensor, or fine orientation sensor.

- **Motivation:** *Derived. Given by requirement 5.1.6.2.*
- **V&V:** *Test. Compare data from flight to fault indicated by ground station.*

5.1.7. The fault management system shall respond to a user recovery command from the ground station unit by power cycling the primary and secondary reaction wheels, coarse orientation sensor, and fine orientation sensor.

- **Motivation:** *Aligns with standard operating procedure in industry today.*
- **V&V:** *Test. Each component can be tested to see if the primary and secondary reaction wheel, coarse sensor, and fine sensor can be power cycled by command from ground station.*

4. Key Design Options Considered

A number of subsystems are necessary to satisfy system requirements, requiring consideration of possible integration solutions as well as individual subsystem solutions.

4.1. TestTable

There were three design options considered for the TestTable: an air table, an ice table, and air bearings. While they are all capable of satisfying Functional Requirement 1, each have pros and cons that were explored to aid in making a design decision between the three of them.

4.1.1. Air Table

The air table option will function in essentially the same manner as an air hockey table. An air source pressurizes a chamber underneath the table surface, which has small, uniformly-spaced holes in it that allow air to flow quasi-uniformly through the table surface. This air flow creates a cushion between the table surface and the underside of the MockSat, forcing the MockSat to lift slightly off of the air table surface, drastically reducing friction. In order to operate as intended, both the surface of the TestTable and the underside of the MockSat must be as flat as possible.

This method is relatively simple to model and manufacture. The number, pattern, and diameter of the holes in the table as well as the volumetric flow rate of the air source are design decisions based on the size and weight of the MockSat and can be tailored to meet the needs of the MockSat.

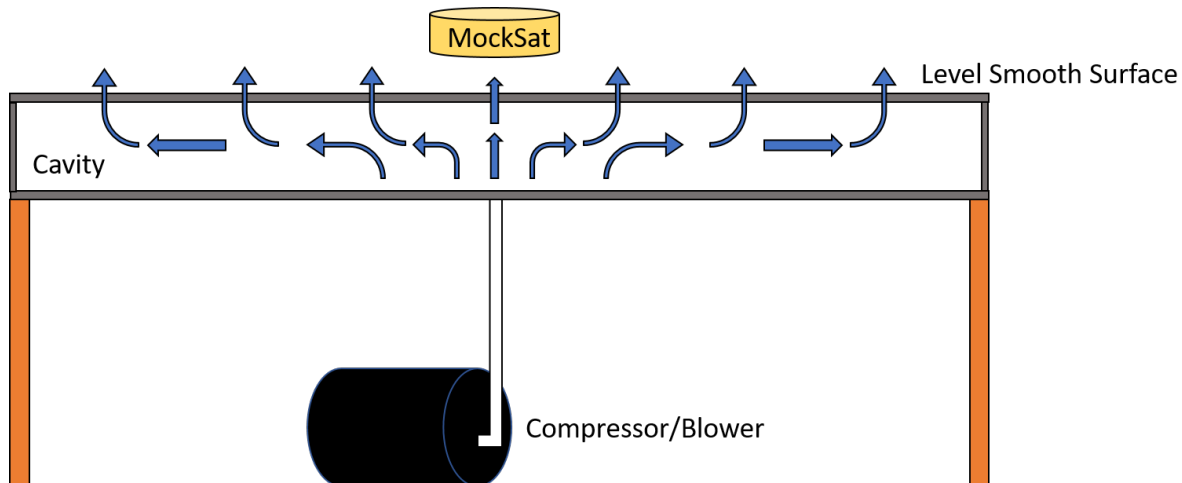


Figure 3. Diagram of air table

4.1.2. Ice Table

The ice table is a flat surface of ice that the MockSat rests on, and since there is always a very thin layer of liquid water on solid ice (on the order of a few molecules), the MockSat has a reduced-friction surface to glide over.

This method places additional logistical and practical constraints on the design of the MockSat. The electrical components of the MockSat must be made water resistant, and the cold operating environment is another factor that must be considered. This design is a logistical challenge, as the table must be frozen solid and surfaced before each test.

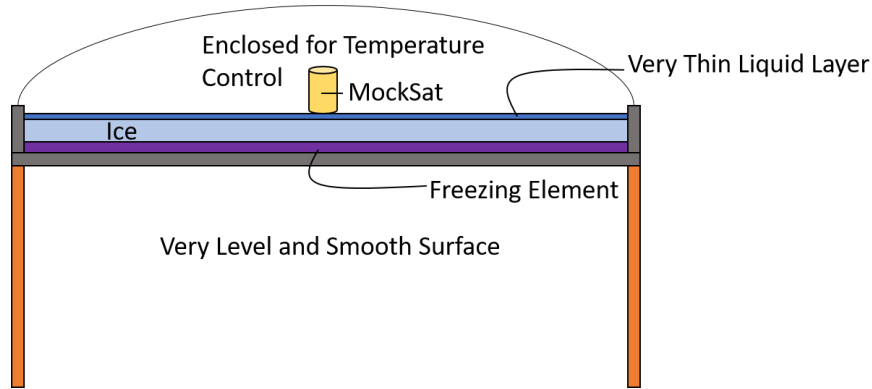


Figure 4. Diagram of ice table

4.1.3. Air Bearings

Air bearings operate by forcing pressurized gas through porous, puck-like structures in order to create a thin cushion of gas underneath them. This cushion of gas supports the weight of the MockSat and provides for a reduced-friction surface.

The air bearings will be mounted to the MockSat itself; a minimum of three are required in order to establish a plane, and will need to be mounted around the perimeter of the MockSat in such a way as to preserve a neutral center of mass. Pressurized gas is provided by an on-board high pressure (HP) reservoir, the size of which will be an engineering decision based on the required run time and the size/weight requirements of the MockSat. On-board supply is required because the dynamics of the MockSat must be isolated in order to comply with Project Requirement 1.1.

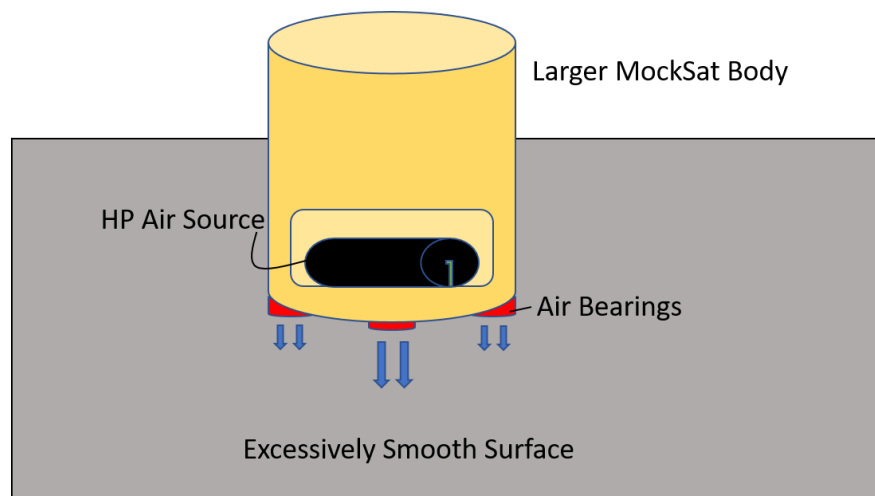


Figure 5. Diagram of air bearing MockSat

Table 2 below gives a quick overview of the resulting pros and cons of each design option.

Table 2. TestBed Pros and Cons

| | Pros | Cons |
|-------------|---|---|
| Air Table | <ul style="list-style-type: none"> -Provides low friction environment by creating an air cushion under the MockSat -Supporting air is provided by the table, no tanks on board MockSat (weight) -Ease of manufacturing -Naturally suited to planar motion -Simple to model -Heritage project to use as resource | <ul style="list-style-type: none"> -Testing area limited to table -Steady air supply and power supply -Very level surface |
| Ice Table | <ul style="list-style-type: none"> -Provides for low friction environment by creating a thin layer of liquid water under the MockSat -Supporting water is provided by the slow melting of underlying ice sheet (thickness will determine testing time) -Naturally suited to planar motion -Relatively simple to model | <ul style="list-style-type: none"> -Requires the electronics of the MockSat to be water-resistant -Requires a sub-freezing storage area large enough to accommodate the table while laying horizontal -Testing to be conducted in a cold environment, lest the table melt at a higher rate -Complicates manufacture by requiring drainage and a method to level the ice sheet once frozen |
| Air Bearing | <ul style="list-style-type: none"> -Provides for a low friction environment by creating an air cushion under the air bearings attached to the MockSat -COTS air bearings available -Relatively simple to model | <ul style="list-style-type: none"> -Supporting air is provided by HP air tanks on board the MockSat -Requires an exceedingly level surface -Three air bearings are required to establish a plane of motion |

4.2. Bounded Translation

Bounded Translation for the MockSat means no control inputs will be induced into the MockSat to control its relative position within the TestTable. The MockSat will be placed in the middle of the TestTable and will be free to move with the desired 3 degrees of freedom, but will be bounded within a defined area of the TestTable. This implies that the full dimensions of the TestTable will be used as bounded area of a smaller bounded area that lies within the TestTable. This is an important factor due to the implications on how it will impact the hardware and dynamics of the MockSat, as well as the management of the injected fault into the MockSat components. To start, Table 3 below shows a pros and cons list identifying important considerations in determining which bounded area (full breadth of the TestTable, or a smaller area of the TestTable) should be used.

Table 3. Test Area Size Pros and Cons

| | Pros | Cons |
|-----------------|--|---|
| Large Test Area | <ul style="list-style-type: none"> -MockSat will traverse for longer period of uninterrupted time. -Fewer collisions with bounded area. | <ul style="list-style-type: none"> -Higher accumulated velocity over longer undisturbed time. -Reaction wheel will saturate fast. -Higher cost to line perimeter of TestTable. |
| Small Test Area | <ul style="list-style-type: none"> - Reaction wheel will not saturate as quickly. -Smaller disturbances induced on MockSat due to small translational area. -Cost less to bound area. | <ul style="list-style-type: none"> -Higher frequency of collisions with bounded area. -Difficulty overcoming constant disturbances which could result in inaccurate attitude determination. |

4.2.1. Bounded Translation

A large bounded area allowing the MockSat to traverse for longer periods of time will align well with the 2D translational control aspect of this project. This will result in fewer disturbances being impinged on the MockSat. This will be a key advantage as the MockSat collides less often with the boundary when compared to a smaller bounded area, therefore, the reaction wheel will have fewer external disturbances to overcome. The biggest downfall of a large bounded area is that the MockSat will have uncontrolled translation for longer periods of time so the reaction wheel will have to account for this. This means that the reaction wheel could saturate more quickly compared to a test on a smaller bounded area.

A small bounded area has several advantages, of which, the MockSat will have a smaller range of motion, and if collisions occur, the MockSat will have less velocity. The downside of this is the potential of having a higher frequency of collisions if there are variations in the testing surface. With the smaller range of motion, the reaction wheel will not saturate as quickly throughout the testing time of 60 minutes. Another advantage is that the cost to implement a small bounded area within the TestTable will not be nearly as high compared to the larger area. In conclusion, the choice of a smaller bounded area outweighs bounding the MockSat to the entire breadth of the TestTable. The main reasoning behind this is that no matter what decision is made as to how the area is bounded, the cost will be less, and the reaction wheel will not have to account for the induced rotation from the TestTable.

4.2.2. Diagrams and Description of Bounded Area

For the magnetically bounded area for the TestTable, a pictorial representation of how this will be done is seen below in Figure 6. Placing a magnet on the MockSat and a similarly poled ring of magnets around the desired bounded area. The magnetic force field from both of these magnets will give a bound to the two-dimension translation. The success of this implementation will be highly dependent on the precise placement of the magnets on the MockSat and the bounded area. Any unaccounted-for misplacement of the magnets could result in undesired torques to the MockSat from the magnetic field. There are two main options for magnets: neodymium magnets and electromagnets. Neodymium magnets are very strong and are manufactured to be small and compact, which will be useful in implementing them with the MockSat. Electromagnets are a viable option because the strength of the magnetic field can be varied to form fit the desired bounded area of the MockSat.

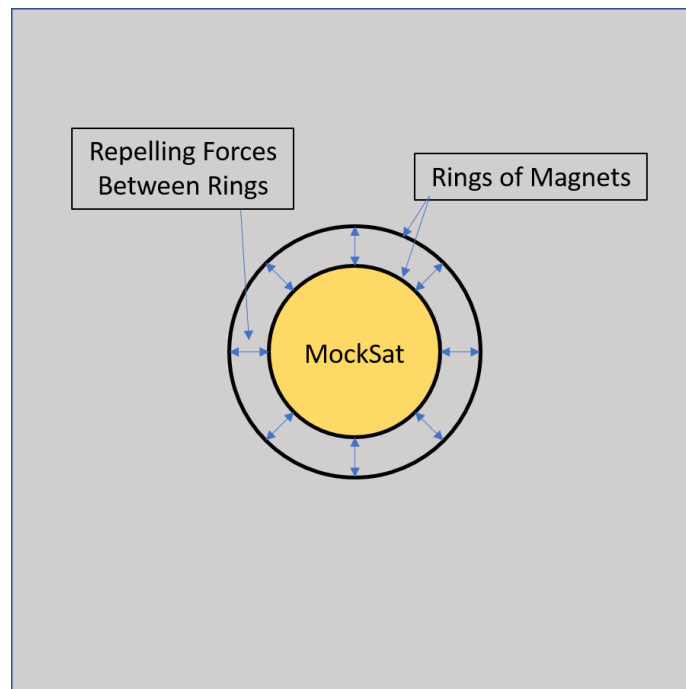


Figure 6. Diagram showing use of magnets for bounding translation

Below, Figure 7 depicts the design possibility of using compression springs attached to a collar which will give the initial bounds for the MockSat. The collar will be connected to the MockSat through bearing that will allow it to rotate freely, as well as small freedoms in two degrees of translation, therefore allowing for pointing accuracy of the sensors to be stable. This design choice brings unique advantages that other options don't such as the spring constant can be chosen to be very low so that when the MockSat pulls on a spring, there will be a cushion to mitigate the translational velocity by the tension difference on the other three.

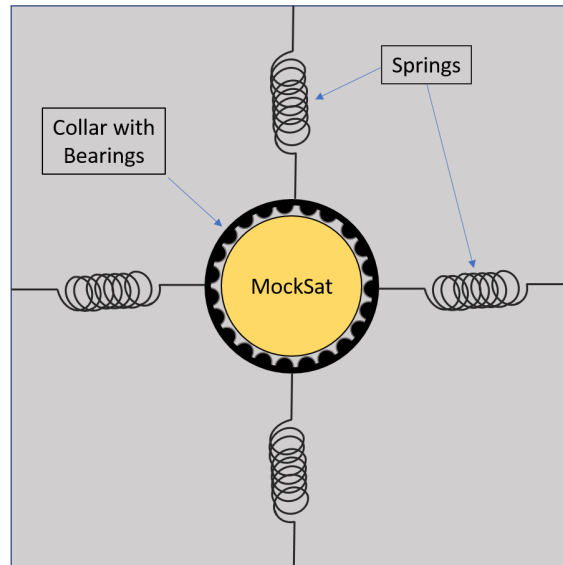


Figure 7. Diagram showing use of springs and a collar with bearings for bounded translation

Figure 8 illustrates a rigid wall that bounds the MockSat within a certain area. This is not a very viable option due to the fact that, when the MockSat collides with the rigid wall, there is no dampening aspects that will help the MockSat stabilize within the bounded area. Due to this, the MockSat will be bouncing around the inside of the bounded area, which will severely impact the rotation and sensors within the MockSat.

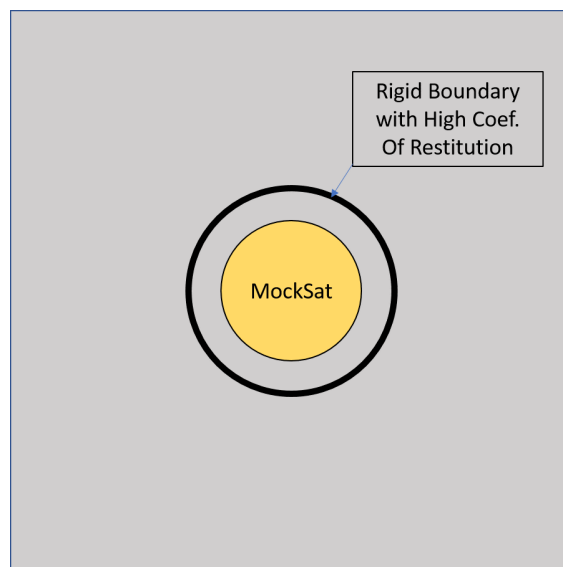


Figure 8. Diagram showing use of a rigid boundary for bounded translation

The last option for bounding the MockSat within a fixed area within the TestTable is using a string attached to the top of the MockSat. The cost of this option is very low due to the fact that it is essentially a string/wire that bounds

the MockSat. The implementation of this option is also easy because it would just require a boom over the top of the MockSat that the string/wire can be fixed in place. The drawback to this option is that there could be uncontrolled pendulum dynamics that could cause difficulties with the pointing accuracy and rotation of the MockSat.

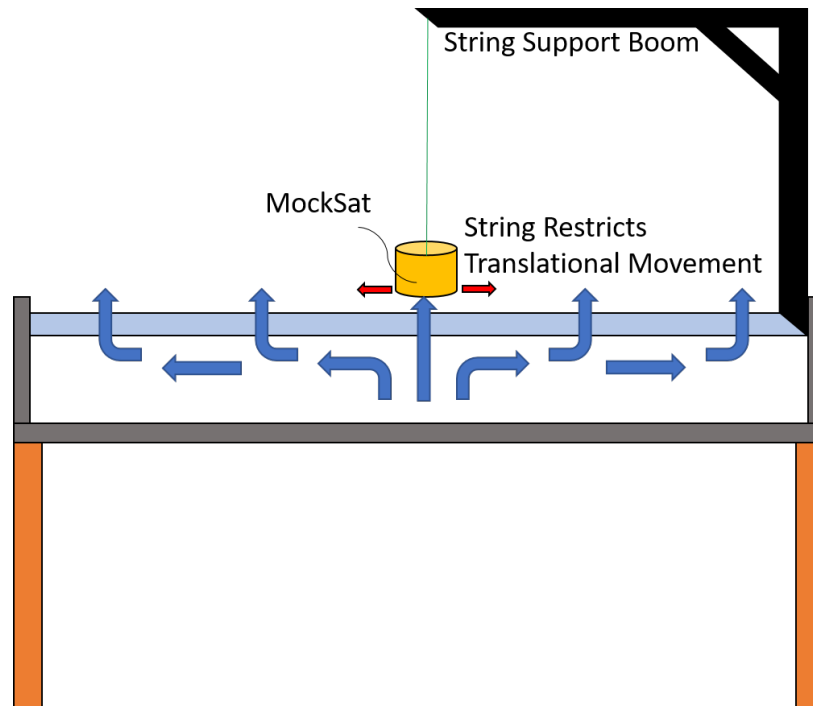


Figure 9. Diagram of string support system for translation

Section 5.2 will go into detail on the trade study of these various options.

4.3. Attitude Determination Sensors

There are numerous sensor options for determining attitude. All such options can be classified as either absolute or relative. Absolute attitude sensors determine attitude directly (i.e. sun sensors, star trackers, magnetometers, etc.) by comparing to some known reference point; relative sensors accomplish the same task by sensing change and integrating to get attitude (IMU, rate gyros, accelerometers, etc.). Most satellites, including GOES-16, use an array of both absolute and relative sensors in order to accurately determine its attitude; as a result, this system will utilize both types of sensors as well. The GOES-16 attitude sensor array is highly complex, but can be simplified down into coarse and fine absolute sensors, and relative sensors; to nominally replicate the GOES-16 ADCS, the MockSat will be equipped with at least one of each.

4.3.1. Relative Sensors

As mentioned above, relative sensors measure change and integrate in order to get attitude data. In practice, there is a very limited number of options. These are:

- Rate Gyros - rotation
- Accelerometers - translation
- Inertial Mass Units (IMUs) - rotation and translation

All of these are comparable in accuracy, attainability and price. In order to reduce the number of components required, this project will utilize IMUs. An IMU meets all requirements for the MockSat and allows for translational sensing for future generations of the MockSat.

4.3.2. Absolute Sensors

The absolute sensors are references fixed to some physical parameter. These sensors typically fall into two categories: optic and non-optic. Optic sensors use some portion of the electromagnetic spectrum, whereas non-optic sensors reference some other constant physical reference. Due to limitations on size and how accurately the sensor system would model the GOES-16 configuration, only the optic sensors were thoroughly investigated. The types of optic sensors and the pros and cons of these sensors can be found in table 4

Infrared (IR) Sensor

The use of infrared (IR) wavelengths is a highly common practice in the development of satellites. It is implemented by the use of contrasting a reference IR source, often the Earth or the Sun, against a background, using that contrast to determine orientation. This effect could be replicated using small-scale IR sources mounted at a desired location and a thermal camera mounted on the MockSat. Depending on the resolution of the thermal camera, IR sensors are suited for both coarse and fine attitude determination. However, due to the large jump in cost for resolution and range in a thermal camera it is unlikely that IR would be used for the fine level of resolution.

Quick Response Code (QR) Sensor

Quick response (QR) codes, like the one seen in Figure 4.3.2, are popular for a broad range of applications. Much like traditional bar codes, QR labels encode information to a physical data matrix that can be read by machines. There are numerous QR Reader applications available for download on modern smart phones. The application for this project are as follows.

A QR code would be the attitude reference point for the MockSat. The MockSat would be equipped with a camera able to recognize and scan the QR code once it came into view. Once the code was scanned, image processing would evaluate where the QR code was located in the picture, and determine maneuvers needed to center the QR code in the camera's field of view. Note the center of the camera field of view is analogous to the pointing vector of the MockSat. The required maneuver information would then be fed to the MockSat's ADCS system to be executed.

There are a number of pros to this attitude method. First, the QR code is very recognizable and therefore not prone to background interference in the image. Once the code is scanned, the QR code has multiple points of reference (i.e. the many black and white squares) allowing the image processing to use averaging techniques to better determine the MockSat attitude. It is for this reason the QR sensor would most likely serve as the fine sensor on the MockSat. As mentioned above, QR codes are very popular, and as a result there is an enormous amount of available literature, legacy code, etc. The major drawback of using a QR sensor is that the required image processing is very computationally expensive.



Figure 10. Example QR code

Visual Point Source (VPS) Sensor

Much like the QR sensor, a VPS sensor utilizes a camera to look for a reference physical shape. In general, this can be anything, but in practice it is typically a simple solid color such as a 6 inch red dot. As a result, this type of sensor is directly analogous to the sun tracker found on the GOES-16. There are two major advantage of the VPS over the QR sensor: the first is it is less computationally expensive to recognize and process a single shape versus a full QR code; second, it can be recognized at lower resolution (cheaper camera and/or larger distance). As a result, VPS is most naturally suited to be the coarse attitude sensor on the MockSat. Once possible configuration would be to mount a fish-eye lens camera on top of the MockSat and place the the reference point on above the boarder of the TestTable. With this setup, the reference point would be withing the fish-eye camera field of view in any orientation. Another configuration would be to use a wide angle lens and program the ADCS system to rotate until the VPS comes into view. Although it is more computationally expensive, the VPS method could be used as a fine attitude sensor as well. This would involve a narrow angle camera with a high zoom, referencing some small VPS in a similar fashion to the coarse sensor.

Lateral Effect Photodiode Sensor

A lateral effect photodiode (LEP) is a membrane that can sense where a beam of light strikes it. It does so through the use of a light-sensitive membrane that generates an electrical signal depending on the angle at which the light strikes it. LEPs are high precision scientific instruments and offer the highest resolution of all options considered for the MockSat. In order to maintain this high level of accuracy, the field of view and range of distances able to be detected are rather small. As such, they are best suited as an option for the fine sensor.

Table 4. Level 1, 2, and 3 Objectives

| Sensor Type | Pros | Cons |
|-------------|---|--|
| IR | (1) minimal interference from background, (2) close analog to sun sensor | (1) Limited sensor-target operational distance, (2) high resolution to price correlation, (3) wide angle lenses not easily/cheaply available |
| QR | (1) above mount combines translational and rotational reference, (2) possibly combines coarse and fine absolute attitude sensors, (3) extensive pre-existing literature, (4) high fidelity attitude sensing | (1) requires higher resolution cameras, prone to fail due to image distortion from wide angle lens |
| VPS | (1) works with virtually any recognizable reference point, (2) not inhibited by wide angle lens distortion, (3) cheapest and simplest option | (1) less accurate, (2) prone to background interference |
| LEP | (1) most accurate option, (2) low computational requirement | (1) limited field of view, (2) expensive, (3) requires special handling |

4.4. MockSat Processor

The will be an integral component of the MockSat. The two choices available to this project are a Field Programmable Gate Array (FPGA) and a microcontroller. The primary benefit of the FPGA over the microcontroller is it's high clock speed and flexibility. However, there is little experience with FPGA's internally, among the team. A microcontroller is desirable due to the teams's existing experience with its development.

Table 5. FPGA vs Microcontroller

| | Pros | Cons |
|-----------------------|--|--|
| FPGA | - Extremely Flexible - High speed signal processing | - Steep learning curve - Very little team experience with HDL |
| Microcontroller Board | - Built in memory - Built in I/O ports - Team Experience - Resources within CU Aerospace Department | - Limited processing capabilities |

With the decision of using a microcontroller, three options were researched for capabilities that will be applied to the project. Since all of the microcontrollers in consideration are roughly the same cost (35-50 USD) cost was not highly weighed in the consideration of the MCU. Some of the most important capabilities of the microcontrollers are outlined in the table below. When considering a microcontroller, the team is primarily concerned with the number of peripherals (GPIO), DAC/ADC capability, and the amount of internal flash memory. A high number of GPIO ports is desirable given the possibility of continued development with regards to future projects and the sensors or actuators that may be required. The ability for DAC/ADC conversion is weighed heavily into the MCU used on board the MockSat. This stems from the high possibility of using analog sensors, such as the lateral effect photodiode and slit camera. Finally, the memory on board the processor should be plentiful enough to allow for the many software packages (flight software, fault management/injection software). The team has limited experience with development of highly intensive

software packages, therefore having a low restriction on the memory available is desirable. Additionally, flexibility regarding the development languages (not being limited to only C/C++) was requested by the software development team. Finally, the capability regarding floating point computation is desirable given the high level of precision required for meeting the pointing accuracy objective.

Table 6. Microcontroller Technical Specifications

| MCU | CLK (MHz) | GPIO | ADC | DAC | Comms | Memory | Languages | Addtl. Pros |
|----------------------------|-----------|------|------------------|---------------|--|---|-------------------------|---------------------------------|
| ATSAME70Q21 ² | 300 | 114 | 2x 12 bit 24 ch. | 12 bit 12 ch. | 5 SPI 3 I2C 5 UART 3 USART USB 2.0 | 2Mb Flash 384 Kb RAM 16 Kb ROM 16 Kb Cache | C/C++ Python JAVA | FPU 1 Comparator Ethernet |
| PIC18F87K22 ³ | 64 | 69 | 12 bit 24 ch. | Addon | 2 USART I2C 3/4 SPI | 128 Kb Flash 4Kb RAM | C Basic Pascal | ASEN 5519 3 Comparators |
| Arduino Due ^{4,5} | 84 | 54 | 12 bit 16 ch. | 12 bit 2 ch. | 1 UART 3 USART USB TWI SPI | 512 Kb Flash 96 Kb RAM | Arduino (C/C++) | ASEN 3300 |

4.5. Fault Injection System

The fault injection system is a critical subsystem of the fault management test bed. Below, 4 design configurations are considered below. Since the fault injection system is closely tied to the ADCS system, each configuration is shown on a high-level ADCS control FBD. This setup well describes each system’s relationship to the rest of the ADCS design, and allows for consistent comparison between each configuration.

The MockSat setup will have at least one micro-controller unit (MCU) to house the flight software and fault management. As part of the following analysis, it needs to be determined whether to use a separate and secondary micro-controller unit (MCU) for the fault injection system. Fault management is tied in with the ADCS but is not shown in Figures 11-14 to demonstrate solely how the fault injection can fit in to ADCS.

In Figures 11-14, there are several common elements: Des, Ref, Control Law, Signal Conditioning (S/C), Actuator, Coarse Sensor, Fine Sensor, and Plant. These are elaborated on here so as to avoid unnecessary repetition. Des and Ref stand for desired pointing and reference pointing. The Control Law block describes the controlling algorithm of the ADCS system. The Signal Conditioning block contains the conversion of software commands into electrical signals to control the actuator, and the conversion of electrical signals from the coarse and fine sensors into software values. Often the signal conditioning is handled by the processor firmware. The Actuator is the reaction wheel actuator, and the Plant represents the physical MockSat system. The Coarse and Fine Sensors are self-evident.

What is not shown on these diagrams is the communications system. Since each of the configurations below requires the use of fault injection software to control the injection mechanism, all of them have the ability to utilize the communications capabilities/peripherals of the MCU, therefore communications is not a differentiating factor in the design decision.

4.5.1. Fault Actuation System

This design option for the fault injection system primarily concerns the use of physical fault injection mechanisms to create faults in the reaction wheel, coarse sensor, and fine sensor. An example mechanism would be a brake which closes on the reaction wheel to slow and/or stop the reaction wheel to simulate bearing failure. Should this design configuration be chosen, the further investigation must be done to determine the best actuation for these physical failures and whether there is a design that won’t damage the original flight hardware. The fact that a single physical injection mechanism will create faults in the reaction wheel, coarse sensor, and fine sensor, the system lacks modularity and the potential for modularity in the future unless hardware is added or changed out. Figure 11 displays the FBD for this configuration. In this setup, a second process is needed to control the fault actuators that ultimately cause to fail.

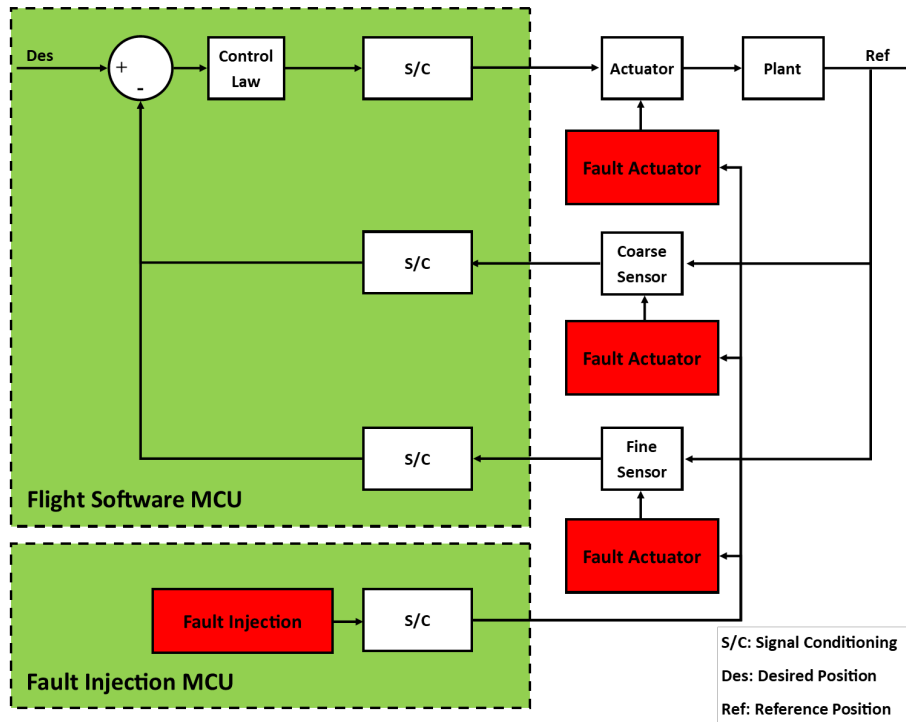


Figure 11. Fault injection through additional fault actuators

4.5.2. Fault Injection Software on Single MCU

Integrating fault injection into the flight software allows for more modularity of the type of faults that can be simulated. This is the one design the solely requires one MCU, making it the cheapest design. Fault injection would act as a filter of data, usually never interfering with flight software, unless the user specifies a fault to be injected. Also, if the dynamics of the MockSat are known, the flight software and fault injection could be tested and modeled to see if the MockSat would respond unexpectedly.

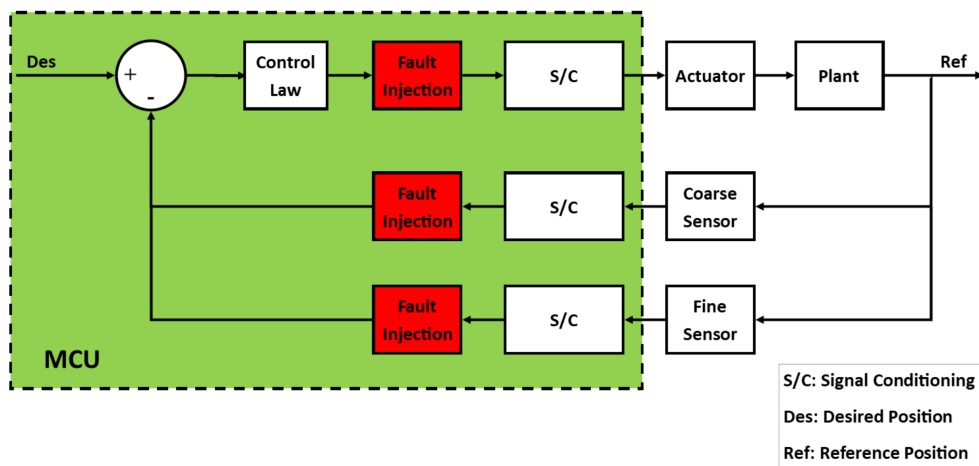


Figure 12. Fault injection lies on MCU with flight software

4.5.3. Fault Injection Software on Separate MCU Option 1

Similarly to fault actuation in the first option, fault injection lives on a separate MCU. However, rather than actuating a fault through some external physical fault injection mechanism, the fault injection is switched on, controlling signals sent to the reaction wheel and back from the sensors. This design could get quite complicated with two softwares living on separate MCUs trying to communicate to the same actuators and sensors.

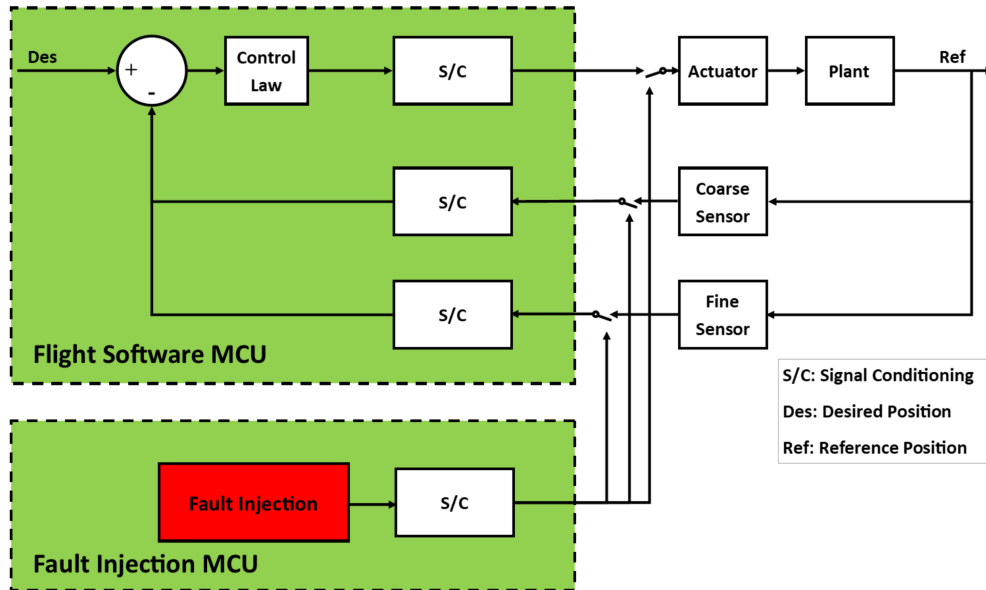


Figure 13. Fault injection lies on separate MCU than flight software and operates on a switch

4.5.4. Fault Injection Software on Separate MCU Option 2

A sort of mix of option 2 and 3 implements fault injection in software, maintaining modularity, but on a separate MCU. Flight software send signals to a separate MCU where fault injection software lives and usually does nothing until the user initiates a fault. Although, it may be more expensive due to the extra MCU, the fact that fault injection is completely separate from flight software and fault management properly mimics how a fault would be injected in satellites such as GOES-R.

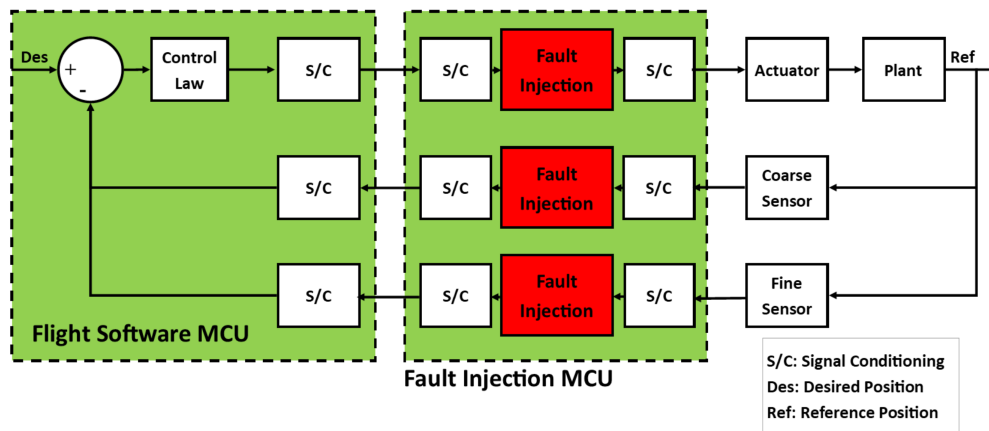


Figure 14. Fault injection lies on separate MCU than flight software and acts as filter

4.5.5. Pros and Cons Table

Table 7. Pros and Cons for Fault Injection System

| | Pros | Cons |
|--------------------------------------|--|---|
| Fault Actuator | -visually verified | -lack of modularity -costly to implement extra hardware -mechanical complexity -interface software with extra hardware |
| Software on Single MCU | -modularity through software -requires no extra hardware -software lives on single MCU | -software complexity |
| FI Software on Separate MCU Option 1 | -modularity through software | -requires two separate MCUs |
| FI Software on Separate MCU Option 2 | -modularity through software | -requires two separate MCUs |

4.6. Fault Management Software

The fault management software will be unaware that a fault injection system exists and will have no interaction with fault injection to properly resemble a true satellite failure. For example, if a single event upset flips a bit, the satellite flight software doesn't have any direct knowledge that an event has occurred without using diagnostic tools. At a high level, there are two main components of any fault management system: Fault detection and fault classification⁶. Upon detection, the fault management software will put the MockSat into a safe mode by turning off the primary and secondary reaction wheels, and alert the user that a fault has occurred, and what type of fault, via communications with the ground station unit. The ground station unit will allow the user to send a recovery command to the fault management system to either switch control to a secondary reaction wheel, power cycle the primary reaction wheel, power cycle the secondary reaction wheel, power cycle the coarse orientation sensor, or power cycle the fine orientation sensor.

For both of detection and classification, the fault management system must have access to data both leaving the MCU and coming into the MCU. The fault management software, then, will live on the same MCU as the ACDS control software. This allows the MCU to pick off data for analysis. Also, being aboard this MCU will enable the fault management software access to the communications system onboard the MCU.

In order to attempt to resolve a fault, the flight management software will also need access to a power conditioning board in order to control power to each of the primary reaction wheel, secondary reaction wheel, coarse orientation sensor, and fine orientation sensor.

The above needs lead to a single self-evident setup for where the fault management system will live, and what it will have access to. Figure 15 shows this setup relative to the control FBD of the ACDS system. This is done since fault management is closely tied and directly impacts the ACDS control system.

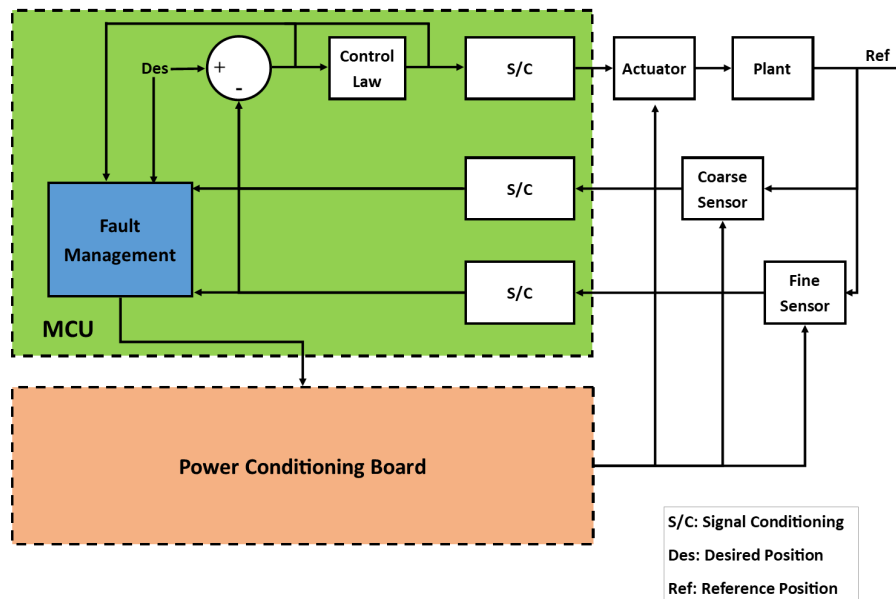


Figure 15. Fault Management With Flight Software

The fault management software gets data streams from the coarse and fine orientation sensors, the desired pointing, and the inputs and outputs of the attitude control law. These data streams will be analyzed by the fault management software to detect and classify a fault. In order to enter safe mode and potentially remove a fault from the system, the fault management system will need to be able to connect with the power conditioning board to control power to the primary and secondary reaction wheels (Actuator in the figure), the coarse orientation sensor, and the primary orientation sensor.

4.7. Battery Types

There are many ways to power a system, but since it is very desirable in this project to have a wireless set up, batteries have been chosen to be the best option. After this, the type of battery to be used must be narrowed down, as there are many different kinds.

4.7.1. Lithium Based Rechargeable Batteries

The first category of batteries analyzed was rechargeable batteries that use lithium. The two main types were Lithium Ion (Li-ion) and Lithium Polymer (LiPo). Both of these types of batteries have comparable specific energies and voltage ratings, and are similar in a few ways. They both are generally available in higher capacities, are readily available, and are lighter than nickel based batteries. The biggest downside to rechargeable lithium batteries are their relative level of safety. Both types are prone to puncture, which could cause a fire. Also, each type requires special chargers, since it is dangerous to over-charge them. Also, they must not be discharged to low, or at too high of a current, since this could also cause a dangerous situation. Despite this, both types of batteries are used widely today, and with proper precautions can be very safe and effective.^{7 8 9 10}

4.7.2. Nickel Based Rechargeable Batteries

The next category of batteries analyzed were nickel based rechargeable batteries. The two researched for this type were Nickel Cadmium (NiCd) and Nickel-Metal-Hydride (NiMH). Both of these types of batteries have lower specific energy and voltages than lithium based batteries. Some positive properties of these batteries is that they are readily available in standard forms like AA or AAA, which could make them easier to implement in a system. It is also a good thing that they are easily rechargeable. A major downside though is that these types of batteries tend to have a high self-discharge rate, meaning they lose charge if they are just sitting and not being used. Also, both of these types of batteries suffer from memory effects, which can cause the battery to not behave at its nominal performance. This can be caused by not fully charging or discharging the batteries with each use, or discharging the batteries too much. Also, NiCd batteries are relatively toxic, and so care must be taken when using and disposing of them. NiMH batteries are less toxic than NiCd.^{7 10}

4.7.3. Non-rechargeable Batteries

For non-rechargeable batteries, the last type looked into, the types researched were standard alkaline and lithium primary batteries. Both of these batteries have high specific energy ratings, and are cheap and readily available at most stores. The biggest downside to them is that they are not rechargeable, which would end up dramatically increasing cost. When used continuously, they are drained much faster, which will require multiple changing of the battery during each test, or purchase of new ones for each test. These batteries are very safe, but are strongly prohibited by the fact that many of them would have to be purchased. This cost would end up outweighing the cost of a few rechargeable batteries and a charger very quickly.¹¹

4.7.4. Pros and Cons Table

Table 8 below contains some parameters for six different kinds of batteries, as well as pros and cons for each.

Table 8. Types of Batteries, Including Specific Energy, Typical Voltage, Pros, and Cons.

| Battery Type | Specific Energy (Wh/kg) | Typical Voltage (V) | Pros | Cons |
|------------------------------|-------------------------|---------------------|---|--|
| Lithium Ion (Li-ion) | 100-250 | 3.7 | -Low self discharge -Generally high capacity -Cheap, readily available -Lighter than Nickel based batteries -Rechargeable | -Prone to over-charging/discharging, puncture -Must have special charger |
| Lithium Ion - Polymer (LiPo) | 100-250 | 3.7 | -Lighter weight than Li-ion -Improved safety over Li-ion -Many possible shapes/configurations -Rechargeable | -Prone to over-charging/discharging, puncture -Must have special charger -More expensive than Li-ion |
| Nickel Cadmium (NiCd) | 40-80 | 1.2 | -Average life cycle -Rechargeable -Standard forms (AA, AAA etc) | -High self discharge -High toxicity -Memory effects |
| Nickel-Metal-Hydride (NiMH) | 60-120 | 1.2 | -Low toxicity -Rechargeable -Standard forms (AA, AAA etc) | -High self discharge -Low life cycle -Memory effects |
| Alkaline | 200 | 1.5 or 9 | -High specific energy -Readily available, cheap -Many familiar forms (AA, AAA etc) | -Low current draw -High internal resistance -Will use quickly during continuous use -Non-rechargeable |
| Lithium Primary | 400 | 1.5 | -Higher specific energy than alkaline -Readily available -Cheap | -Low current draw -High internal resistance -Will use quickly during continuous use -Non-rechargeable |

4.7.5. Power Flow

Regardless of what type of battery is chosen, there will be a set flow of power throughout the system, as shown below in Fig. 16. Power will flow from the battery pack through a power conditioning board containing various DC to DC converters. After the power is converted to the correct DC level for each sensor, it will go through to power each respective part. The one note for this diagram is on the box that says "Memory/Comms": depending on the level achieved, there may be power supplied to either an on board memory storage system, or the system that provides

communications. All of the other systems like micro-controller, orientation sensors, etc. will be necessary for all levels of success. Some of the voltage levels in the diagram below are left as TBD because they will depend on the specific sensor or motor chosen.

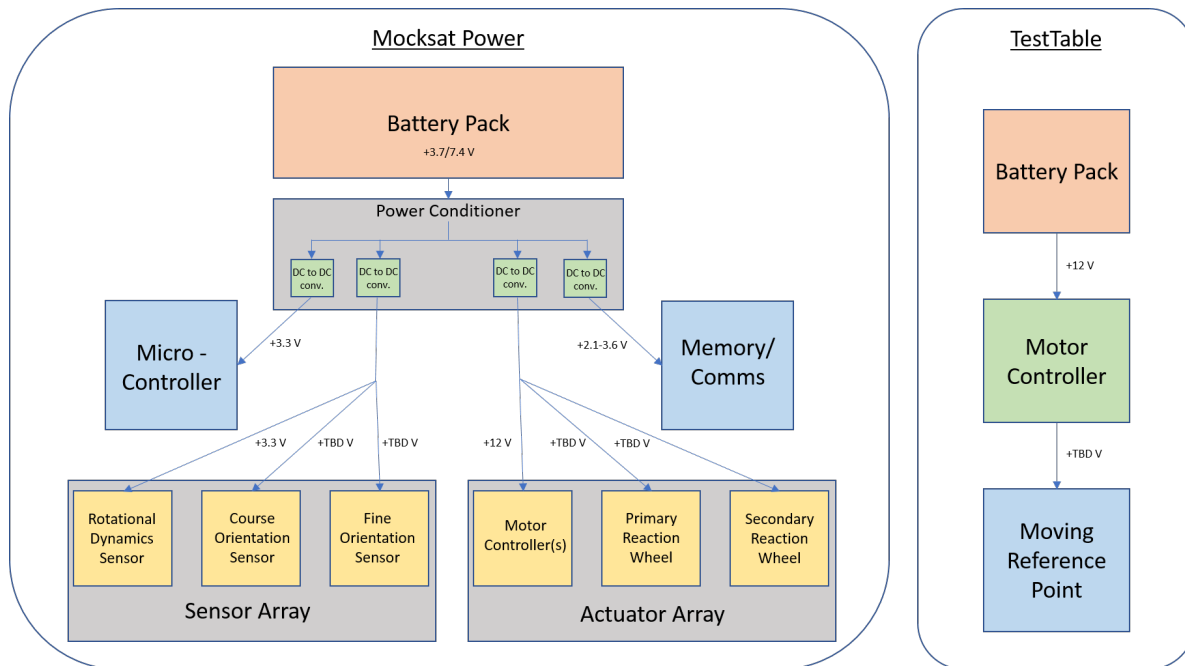


Figure 16. Diagram of flow of power throughout Mocksat and TestTable

4.8. Communications

For the communications portion of the MockSat, we will be designing for the level three goals of the project, meaning that wireless communication will be necessary. This will be accomplished with radio frequency (RF) communication. It has been decided that two transceivers, one aboard the ground station and one aboard the MockSat, will be used. This will be the easiest and most cost effective way to implement wireless communications. The module used will end up being a Xbee or Xbee-pro Zigbee module. These are items that the team has had experience with in a previous course and in work outside of school. There is also a wide use of them throughout faculty projects and research, making it easy to get faculty advice regarding them.

According to the datasheet for these modules¹², they have a range of 200 to 300 feet indoors, and up to 2 miles with line of sight. This is much more than necessary since the MockSat will be less than 10 or 20 feet from the ground station at all times when in operation. Each of these modules have a data transfer rate of up to 250 Kbps. This is orders of magnitude more than what will be transferred during our test. These modules transmit on the 2400 MHz band, and are approved through regulations in the United States.

Regarding the lower level objectives of communication, wired connections will be used. At level one, data will be stored on flash memory that will be removed after the test to analyze this data. At level two, there will be a wired connection that performs the same functions as the wireless one, only over a hard link. As stated, we are designing for level three, so most details will be given for the RF communication. Also, communications aren't an integral part of this project: all of our main objectives could be accomplished if we only achieved level one of communications. This means that when attempting to achieve level three, the main focus for the communication is to acquire low cost, easy to implement RF modules that will allow real time data transmission that won't interfere with the dynamics of the MockSat. This is why there won't be a trade study performed for communications.

5. Trade Study Process and Results

Trade studies are used to evaluate different design options based on how well they satisfy the various functional and technical requirements. The individual categories are varied for each trade study, but one overall scoring scheme is used throughout the trade studies to follow. The most desirable option relative to each category is given a score of "5", and any option that doesn't fulfill the category is given a score of "0". The scoring criteria seen in table 5 was used for the sake of consistency.

Table 9. Trade Study Scoring Criteria

| Score | Criteria |
|-------|---------------------------------|
| 0 | Does not fulfill requirement |
| 1 | Barely fulfills requirement |
| 2 | Marginally fulfills requirement |
| 3 | Fulfills requirement |
| 4 | Fulfills requirement well |
| 5 | Most desirable |

5.1. TestTable

The TestTable trade study was conducted using the following categories:

- **Test duration:** This system will require numerous instances of long testing periods in order to not only properly calibrate the ADCS system but also evaluate the fault injection and fault management systems. This is the most heavily weighted category for this reason. Any TestTable method which lengthens the available testing time is scored higher than an option that inherently restricts the testing time.
- **Cost:** The time and money required to implement the method was weighted based on the relative effect on the project budget and the time demand.
- **Manufacturing Required:** The *amount*, *type*, and *difficulty* of manufacturing and assembly factor into this category.
- **Heritage:** This category is a representation of the access to research conducted for past senior projects and the relative experience the PAB members have with the method.
- **Simplicity:** How simple the method is to model weighed heavily into this category, as well as how simple it is to operate and to what extent the method complicates the MockSat structure.
- **Logistics:** Ease of test setup/breakdown, transportation, surface preparation and maintainability all factor into this category.

Table 10. TestTable

| Criterion | Weight | Air Table | Ice Table | Air Bearing |
|------------------------|--------|-------------|-----------|-------------|
| Test Duration | 35% | 5 | 3 | 1 |
| Cost | 10% | 5 | 3 | 1 |
| Manufacturing Required | 15% | 2 | 5 | 4 |
| Heritage | 5% | 5 | 0 | 1 |
| Simplicity | 15% | 5 | 2 | 1 |
| Logistics | 20% | 5 | 1 | 1 |
| Total | 100% | 4.55 | 2.6 | 1.45 |

The air table method dominated the trade study, as Table 10 clearly shows. The air table scored the highest in all categories save one; the amount of manufacturing required is substantial, since the team expects to design and build an air table from scratch. Using the heritage air table would be beneficial from a cost and scheduling perspective, but it is not yet clear if that air table will suit this project's needs.

The ice table scored relatively poorly due to a combination of logistical challenges, reduced test duration, and increased complexity. The logistics of freezing and resurfacing a large ice sheet prior to each test, the reduced test time resulting from the ice sheet slowly melting, and the need to make the MockSat water-resistant as well as capable of operating in cold conditions served to bring the ice table score down.

The air bearing method scored the lowest in all categories except one, the manufacturing required. This is because the air bearings themselves would be purchased items, and while they need a very smooth and flat surface on which to move (which will need to be created), that is the extent of the manufacturing required to utilize this option. It scored poorly in the Test Duration, Simplicity, and Logistics categories due to the fact air bearings require a steady supply of high pressure gas to operate, and requirement 1.1 states the dynamics of the MockSat must be isolated. A HP gas source must then be carried on-board, which limits the test time, complicates the MockSat structure, and presents a logistical challenge by necessitating the constant removal, refill, and re-installation of HP gas bottles.

5.2. Bounded Translation

The Bounded Translation trade study was conducted using the following categories:

- Cost: The time and money required to implement the method was weighted based on the relative effect on the project budget and the time demand.
- Impact on Pointing Accuracy: The ability and accuracy to point the MockSat at a desired location. An example is the if the area was bounded by the rigid wall, the collisions with the wall will cause inaccuracies and ability to point towards the desired location.
- Implementation Complexity: The difficulty to properly implement a uniform bounded area.
- Rotational Restriction: How the bounded wall will impact the rotational dynamics of the MockSat when collisions occur.
- Settling Stability: The tendency for the MockSat to converge to a final resting space within the bounded area.

Table 11. MockSat: Bounded Translational

| Criterion | Weight | Magnetic Ring | Spring and Bearings | Rigid Wall | String/Wire |
|---------------------------|--------|---------------|---------------------|------------|-------------|
| Cost | 10% | 3 | 2 | 5 | 5 |
| Impact on Pointing Acc. | 20% | 5 | 4 | 1 | 4 |
| Implementation Complexity | 25% | 2 | 2 | 4 | 5 |
| Rotational Restriction | 15% | 5 | 3 | 1 | 4 |
| Settling Stability | 30% | 5 | 5 | 1 | 3 |
| Total | 100% | 4.05 | 3.45 | 2.15 | 4.05 |

The three main aspects from this trade study to take into account are the settling stability, implementation complexity, and impact to pointing accuracy. Settling stability is weighted the highest due to the fact that the bounded area needs to allow for settling as to not impact the pointing accuracy of the sensors on the MockSat. If there is no settling of the MockSat, then the reaction wheel and sensors on the MockSat will constantly have to overcome these impacts. Implementation complexity is rated the next highest due to the fact that whatever option is chosen, the option needs to be implemented in such a way that does not impact the dynamics of the MockSat which will cause a failure of multiple design requirements. Lastly, impact to pointing accuracy is rated at 20% due to the fact that, however the area is bounded, it cannot cause additional complexity to how the MockSat will point to the desired location. Implementing a design that will add complexity to this does not make logistical sense. Another aspect of the trade study, cost, was rated the lowest due to the fact that all of these options would cost less than 100 dollars. Due to this trade study, depicted in Table 11, two options tied as the forerunner of bounding the test area. The two are: a magnetically bounded area and a string/wire that bounds the translational movement.

5.3. Attitude Determination Sensors

The MockSat will be equipped with both a coarse and fine attitude sensor in order to best mimic the sensors found on GOES-16. The coarse sensor will be responsible for getting the MockSat close enough to the correct attitude to allow the fine sensor to take over determination responsibility. The fine sensor will provide the necessary feedback allowing the actuators to orient the satellite to within the specified 5 degree window. Coarse sensor options will be evaluated in the following categories:

- Cost: The price of the sensor and any additional required hardware.
- Accuracy: The ability of the sensor to consistently provide reliable results.
- Data Ease of Use: The processing ease of the raw sensor data.
- Implementation Complexity: How many restrictions it places on the rest of the system and how hard it is to integrate.

There were three options considered for the MockSat. A trade study was conducted and the results are recorded in Table 12.

Table 12. MockSat: Coarse Attitude Determination

| Criterion | Weight | VPS (Fisheye) | Thermal | VPS (Wide Angle) |
|---------------------------|--------|---------------|---------|------------------|
| Cost | 35% | 4 | 2 | 5 |
| Accuracy | 15% | 4 | 3.5 | 5 |
| Data Ease of Use | 20% | 3 | 5 | 4 |
| Implementation Complexity | 30% | 5 | 2 | 4.5 |
| Total | 100% | 4.1 | 2.83 | 4.65 |

It is clear that the Wide Angle VPS was the best option considered for the MockSat coarse sensor. The Thermal option had a number of inhibiting factors. Most notably, the cost of a thermal camera with high enough resolution to meet the MockSat requirements was an order of magnitude greater than the other two sensor options. In addition, the thermal camera would need a very hot source to look for, which has inherent safety problems. The primary advantage of a wide angle lens over a fisheye is the wide angle lens has less distortion in the image. This in turn reduces the amount of required image processing leading to higher refresh rate of the sensing system.

Switching to the fine attitude sensor for the MockSat, there were again 3 options considered: QR, Lateral Effect Photodiode (LEP), VPS. The criteria by which these sensors were evaluated is the same as that for the coarse sensor. The trade study results are recorded below in Table 13.

Table 13. MockSat: Fine Attitude Determination

| Criterion | Weight | QR Sensor | LEP | VPS |
|---------------------------|--------|-----------|-----|------|
| Cost | 25% | 5 | 1 | 5 |
| Accuracy | 25% | 3.5 | 5 | 5 |
| Data Ease of Use | 20% | 3.75 | 5 | 4 |
| Implementation Complexity | 30% | 5 | 1 | 3.75 |
| Total | 100% | 4.38 | 2.8 | 4.18 |

The LEP was initially considered to be the best due to its high accuracy and analog output signal; however its high cost and high implementation complexity make it an infeasible option for the MockSat. The remaining options of the trade study are the QR and VPS, which both scored well and function in similar manners. Both use a visual reference source in conjunction with a narrow angle camera. The VPS had a slight edge over the QR due to its lesser implementation complexity. The QR code requires a clean background, which could be compromised by the coarse reference source.

5.4. Fault Injection System

From Table 7, a trade study is conducted to determine the most effective configuration for fault injection. The criterion considered in the study are as follows:

- Complexity: How complex is the system in terms of hardware, software, and their interface.
- Modularity: The ability of the system to introduce a number of faults.
- Ability to Emulate Real Faults: The ability of the system to replicate faults that have occurred on satellites.
- Price: The relative cost of hardware needed for the system. Specifically the cost of having to MCUs vs having only one MCU.

Both complexity and modularity have the heaviest weights at 30%. Complexity because the team has extensive skills with software, but lacks familiarity with hardware, therefore the design with one MCU is an attractive option because all the software would live in one place. Modularity because the essence of the project lies in the idea the TestTable will allow for a simpler method of testing a number of faults. Using hardware to actuate a fault would limit the possible faults, making it unfavorable for this criterion. The ability to emulate real world faults must be considered heavily at 20% to provide purpose behind the faults being tested. Using hardware to actuate a fault, such as a brake, can replicate real world faults well but software has the ability to replicate multiple faults, whereas hardware will have limitations. Price, of course, must be weighted heavily at 20% because MCU can be expensive and doubling the amount of MCUs will eat in to the budget. For the purposes of the trade study shown in Table 14, 5 is the most favorable option whereas 0 indicates design is completely undesirable as mentioned in Table 5.

Table 14. MockSat: Fault Injection

| Criterion | Weight | Fault Actuation | Single MCU with Fault Injection | Separate MCU with Switch | Separate MCU with Fault Injection Filter |
|--------------------------------|--------|-----------------|---------------------------------|--------------------------|--|
| Complexity | 30% | 1 | 5 | 3 | 3 |
| Modularity | 30% | 0 | 5 | 4 | 5 |
| Ability to Emulate Real Faults | 20% | 4 | 5 | 5 | 5 |
| Price | 20% | 1 | 5 | 2 | 2 |
| Total | 100% | 1.3 | 5 | 3.5 | 3.8 |

5.5. Battery Types

The final trade study performed was to select a type of battery. The results of this can be seen in Table 15. There were five metrics that were considered in this study:

- Price: The relative cost of each type of battery. This takes into account the number of times each type of battery would be purchased, as well as the expense of a charger if necessary. Weighed at 20% since it will be fairly important to keep cost of the battery system low.
- Specific Energy: This is the amount of energy that a battery can hold per mass. This is the most important consideration in this trade study because it will be important to have a lot of power in a fairly small package, since the weight and size of the batteries will affect the rest of the project.
- Rechargeability: This metric is based off of whether the batteries are rechargeable or not, and also includes the relative ease of charging them. Also included here is whether or not the batteries suffer from memory effects related to charging. This is weighted at 20% because it will be important to have batteries that are easy to recharge and use repeatedly.
- Safety: This metric indicates the relative safety of each battery, including possible dangers in charging, puncture, or temperature/storing restrictions. Weighted at 20% since a battery failure could be dangerous and result in fires, making safety a high concern.
- Ease of Use: This metric takes in many factors, including the need to change the batteries part way through the test, the form factor and ease of integration of the battery, and the handling of the battery outside of the test

(storage temperature, disposal, etc.). This is weighted the lowest at 10% because ultimately these factors won't drastically affect the outcome of each test, only cause annoyances outside of the tests.

Table 15. MockSat: Batteries

| Criterion | Weight | Li-ion | LiPo | NiCd | NiMH | Alkaline | Li Primary |
|------------------|---------------|---------------|-------------|-------------|-------------|-----------------|-------------------|
| Price | 20% | 4 | 3 | 5 | 5 | 2 | 2 |
| Spec. Energy | 30% | 4 | 5 | 2 | 2 | 5 | 5 |
| Rechargeability | 20% | 4 | 5 | 3 | 3 | 0 | 0 |
| Safety | 20% | 3 | 3 | 3 | 4 | 5 | 5 |
| Ease of use | 10% | 4 | 5 | 3 | 4 | 1 | 1 |
| Total | 100% | 3.8 | 4.2 | 3.1 | 3.4 | 3 | 3 |

After the study was performed, the top two selections came out to be either Li-ion or LiPo batteries. Ultimately, LiPo batteries will most likely be used because of their similar properties to Li-ion, plus the added benefit of being able to find small form factor batteries of this type.

6. Selection of Baseline Design

6.1. Overall System Design

The trade studies above, coupled with decisions made by the team, will guide the following design of the overall system. The TestTable will be an air table, and will provide an environment which allows the MockSat to move in the specified 3DOF while imposing minimal restrictions on the MockSat's rotational and translational planar dynamics. The ability of the MockSat to translate will be externally limited via either a string tether or a repulsive ring of magnets, with the final design decision to be made upon further research. The apparatus for restricting the translational motion of the satellite will be removable, so as not to prohibit future expanded projects.

The TestTable will also support a point of reference that will be visually detected via a coarse sensor for general pointing and a fine sensor for precision pointing. Information from these sensors will provide data to the flight control software to determine the direction of actuation required by the satellite. A set of reaction wheels will provide attitude control of the MockSat. The magnitude of this actuation will then be determined by the flight control software, at which point a command will be sent to the motor controllers to drive the reaction wheels. Relative dynamic sensors, such as an IMU, will provide feedback to the control loop to continually modify the commands sent to the motor controller.

In conjunction with the flight software, the fault software will be monitoring the avionics system to ensure the commands sent from the microcontroller are the same commands received by the motor controller. The fault injection software will operate in a way so that it remains undetected by the flight software and fault management software, but it will have the capability to alter commands from the microcontroller in a way that results in off-nominal pointing performance. Once a fault has been injected in the system, the response from the MockSat ADCS should differ from the expected response. This inconsistency, between the ADCS command and the plant response, will be what the fault management must recognize. Once a fault has been recognized by the fault management software, the MockSat will enter a safe mode and notify the ground station of the MockSat's status. The safe mode is defined as temporarily cutting power to the reaction wheels while maintaining communication with the ground station for sensory data relay. The MockSat will then remain in safe mode and await instruction from the ground station to re-enter operational functionality. Upon confirmation of the system's ADCS integrity, the ground station will relay a command, via radio frequency, to re-enter operational functionality.

6.2. TestTable

The air table was clearly the superior choice based on the results of the trade study. This option will create an environment that provides the longest testing time, making it an attractive option. The logistical challenges posed by the ice table or an air bearings make them unappealing, while the air table is relatively easy to move, setup and operate with a minimum of prep time required. Additionally, an air table was constructed by a previous project which gives

the team a template for future construction, as well as a apparatus to test the suitability of various design choices to the air table environment.

6.3. Bounded Translation

There were two options that proved to be equally viable for the bounded translation solution. These options are the magnetic ring setup and the string/wired setup. As this solution is not a huge driving factor for the overall success of the project, the team will take some additional time to further investigate these options. This will include some modelling of the systems to get a better understanding of implementation and dynamics.

6.4. Attitude Determination Sensors

6.4.1. Relative Sensors

This project will utilize IMUs. An IMU meets all requirements for the MockSat and allows for translational sensing for future generations of the MockSat. Additionally, advances in the technology have made them comparable in price to accelerometers and rate gyros with their additional capability of dynamic sensing both in rotation and translational motion.

6.4.2. Coarse Sensor

The primary considerations for selecting the coarse sensor were cost and implementation complexity. After the trade study, the wide angle lens proved to be the best choice. All three choices have similar computational expense, however, the wide angle lens will have less distortion and be overall more accurate than the fish eye lens.

6.4.3. Fine Sensor

The fine sensor will be a narrow view camera. For this sensor the primary considerations were accuracy and implementation complexity. Although the visual point source was evaluated to be the least accurate option, it still satisfies accuracy required. In addition, the narrow view camera is the least complex sensor to implement, as well as the cheapest option.

6.5. Battery Type

The best choice for battery for this project is lithium polymer (LiPo). This choice was made primarily because of the high specific energy capacity of LiPo as well as rechargeability. Moreover, there are many options for the size and shape of LiPo batteries, allowing the selection to be tailored to this project.

6.6. Fault Injection and Fault Management Software

In this section, the fault injection and fault management designs are described in unison. Since they are so closely tied together, and live on the same MCU, it is reasonable to present both together in the same design configuration. Additionally, both software systems will require access to communications, which is handled by the MCU.

Figure 17 depicts the final selected design configuration for each of the fault management and fault injection systems.

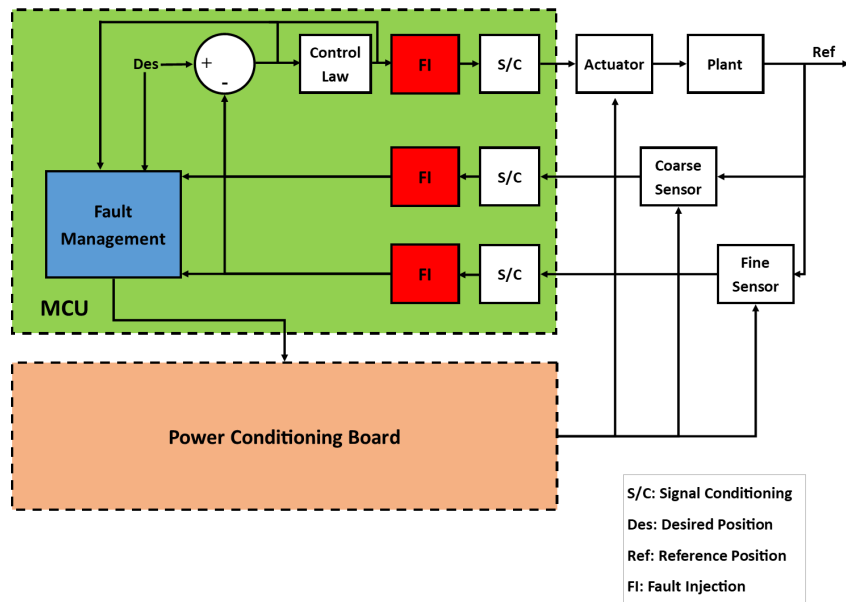


Figure 17. Fault injection and management with flight software

Essentially, the fault injection system will act as a filter and usually will remain invisible unless a ground station operator enacts a fault. Once a fault has been injected, the fault injection software will manipulate the input and/or output data streams to the actuators and/or sensors in order to mimic a fault. Also, the fault injection system will be able to be disabled by the user via the ground station unit.

The fault management system has access to data streams from each of the coarse and fine orientation sensors, desired pointing, and the input and output of the control law. From this dataset, the fault management system will examine residuals to determine if a fault is occurring. Once a fault is detected, the fault management system will put the MockSat in a safe mode state by turning off the attitude actuators via the power conditioning board. Additionally, the fault management system will alert the user that a fault has occurred via the ground station unit, and await user command from the ground station unit. To enable recovery, the fault management system will have the ability to cycle power to any of the orientation actuators or sensors through a power conditioning board.

6.7. MockSat Processor

The microcontroller chosen for MockSat is decided to be the Atmel SAME70Q21. This processor was recommended specifically by a graduate student in the aerospace department. The advice from this individual was heavily considered provided he has had years of experience with embedded systems, as well as experience writing the software for the ADCS system of the 3U cubesat, Polarcube. Observing the technical specifications of the three choices and their equal affordability, the ATSAME70Q21 is clearly superior in every aspect heavily considered by the team. Finally, based on the design selection of integrated fault injection and flight software, the capabilities of the Atmel processor makes it an obvious choice.

6.8. Communications

As discussed in section 4.8, a decision was made to use a radio frequency transceiver like a Xbee or Xbee-pro Zigbee for communications. These modules are cheap and easy to implement, and have a much greater range and data transfer rate than will be necessary. Also, there are many resources online and within the faculty that will be useful in implementing these.

References

- [1] N. Goddard, "Goddard Space Flight Center Rules for the Design, Development, Verification, and Operation of Flight Systems." http://www.goes-r.gov/syseng/docs/GSFC_STD_1000.pdf, August 2014.
- [2] Atmel, "Sam e70," technical datasheet, Atmel, Jan 19, 2016.
- [3] MICROCHIP, "Pic18f87k22 family," technical datasheet, Microchip Technology Inc., 2009-2011.
- [4] Arduino, "Arduino Due." <https://store.arduino.cc/usa/arduino-due>, Feb 2017.
- [5] D. Erickson, "Arduino Due Projects." <http://www.djerickson.com/arduino/>, Feb 2013.
- [6] S. Montenegro and W. Hu, "Detection of actuator faults for an attitude control system using neural network," world academy of science, engineering and technology, International Journal of Mechanical, Aerospace, Industrial, Mechatronic and Manufacturing Engineering Vol:4, No:11, 2010, 2010.
- [7] B. University, "Comparison Table of Secondary Batteries." http://batteryuniversity.com/learn/article/secondary_batteries, April 2016.
- [8] B. University, "How do Lithium Batteries Work?." http://batteryuniversity.com/learn/article/lithium_based_batteries, July 2017.
- [9] B. University, "Is Lithium-ion the Ideal Battery?." http://batteryuniversity.com/learn/archive/is_lithium_ion_the_ideal_battery.
- [10] epec, "BATTERY CELL COMPARISON." <http://www.epectec.com/batteries/cell-comparison.html>.
- [11] B. University, "Advantages of Primary Batteries." http://batteryuniversity.com/learn/article/primary_batteries, April 2017.
- [12] DIGI, "DIGI XBEE AND DIGI XBEE-PRO ZIGBEE." http://www.mouser.com/ds/2/111/ds_xbee_zigbee-709350.pdf.