

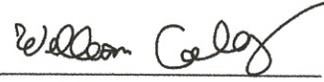
Design of a NanoSat for an Ionospheric Mission

A Major Qualifying Project Report
Submitted to the Faculty of the
WORCESTER POLYTECHNIC INSTITUTE
in Partial Fulfillment of the Requirements for the
Degree of Bachelor of Science
in Aerospace Engineering

by



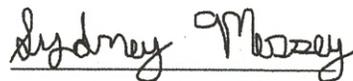
Alia Brown



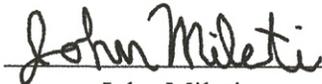
William Cooley



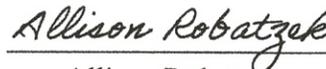
Alexander Klenk



Sydney Messey



John Mileti



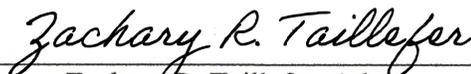
Allison Robatzek



Matthew St Jean

March 18, 2021

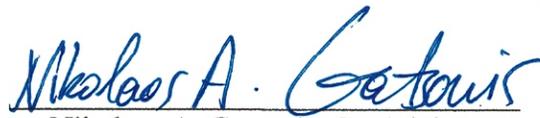
Approved by



Zachary R. Taillefer, Advisor

Assistant Teaching Professor

Aerospace Engineering Department



Nikolaos A. Gatsonis, Co-Advisor

Professor

Aerospace Engineering Department

Abstract

This work presents the design of a 6U nanosat carrying a NASA-designed miniature Ion Neutral Mass Spectrometer (mini-INMS) and the design and construction of a Helmholtz cage. The nanosat is designed to measure particle composition of the ionosphere's F layer by flying on an elliptical orbit with an initial perigee of 250 km and apogee of 400 km. The nanosat will be deployed in a Sun-synchronous orbit and transfer to a lower orbit using an electrospray thruster for orbit raising and drag compensation to extend the mission lifetime which was analyzed in Systems Tool Kit (STK). A novel spinning motion is utilized to expand the particle collection capability of the mini-INMS. Attitude control was provided by magnetorquers with attitude determination and control schemes developed in MATLAB. Mechanical, thermal, and telecommunications analysis is performed using SolidWorks, ANSYS, COMSOL, and STK. Environmental factors, including magnetic field interference, radiation dosage, and particle impacts are also analyzed and accounted for using STK. Electrical power analysis was performed using STK and MATLAB to create a model of CubeSat power generation, storage and use. The Helmholtz cage, composed of three pairs of orthogonal square coils, was designed to simulate the magnetic environment experienced by the nanosat to provide ground-based testing of the magnetorquer-based ADCS. This cage was designed, manufactured, and tested to verify its accuracy. A controllable power source and data collection methods were also constructed to allow many different magnetic environments to be simulated.

"Certain materials are included under the fair use exemption of the U.S. Copyright Law and have been prepared according to the fair use guidelines and are restricted from further use."

Acknowledgements

The team would like to thank the following individuals for their support in completing this project.

Project advisors

Professor Taillefer

Professor Gatsonis

Additional Support

Adriana Hera

Tina Stratis

Ian Anderson

Table of Authorship

1 Introduction	Brown
1.1 Background and Literature Review	
1.1.1 CubeSat Applications and Missions	Brown
1.1.2 Review of CubeSats at WPI	St Jean
1.1.3 Helmholtz Cage	Mileti
1.2 Goals	Robatzek, Messey
1.3 Objectives, Approach, and Standards	All
1.4 Mission Operations and Constraints	Brown
1.5 Design Requirements and Constraints	All
1.5.1 CubeSat	
1.5.1.1 Dispenser background and selection	Cooley
1.5.1.2 Launch vehicle background and selection	Cooley
1.5.1.3 Mechanical	Cooley
1.5.1.4 Propulsion	Messey
1.5.1.5 Power	St Jean
1.5.1.6 Telecommunications	Robatzek
1.5.1.7 ADCS	Klenk
1.5.1.8 Environmental Effects	Brown
1.5.1.9 Thermal Control	Mileti
1.5.2 Helmholtz Cage	Klenk
1.6 Project Management and Budget	Messey
1.7 Task Breakdown and Timetable	Messey
1.8 Facilities and Equipment	Mileti
2 Payload and Design Implications	Brown
3 Mechanical Design and Analysis	Cooley
4 Propulsion	Messey
5 Power System Design and Analysis	St Jean
6 Telecommunications Design	Robatzek
7 Attitude Determination and Control Subsystem	Klenk
8 Space Environment and Effects	Brown
9 Thermal Control Analysis	Mileti
10 Helmholtz Cage	
10.1 Helmholtz Cage Overview	Klenk
10.2 Helmholtz Cage Development and Design	Klenk
10.3 Structural Design	Cooley, St Jean
10.3.1 Magnetorquer Mounting Method	Brown
10.3.2 Power Supplies	Messey, Robatzek
10.3.3 Motor Controllers	Mileti
10.4 Magnetorquer Test Article	St. Jean, Klenk
10.5 Data Collection Methods	Brown, Cooley
10.6 Data Management	Cooley
10.7 Assembly	St. Jean, Messey

10.8 Testing	Klenk
11 Conclusion and Recommendations	
11.1 Social Implications of CubeSats	Brown
11.2 Recommendations	
11.2.1 Spacecraft Charging	Brown
11.2.2 Integrated System	Cooley
11.2.3 Magnetorquers and Spinner Satellites	Cooley
11.2.4 Magnetorquer Test Article Stand	Brown
Appendices	
Appendix A	Cooley
Appendix B	Cooley
Appendix C	Messey
Appendix D	Messey
Appendix E	Robatzek
Appendix F	Klenk
Appendix G	Klenk
Appendix H	Klenk
Appendix I	Klenk
Appendix J	Klenk
Appendix K	Mileti
Appendix L	Klenk
Appendix M	Klenk
Appendix N	All

Table of Contents

Abstract.....	2
Acknowledgements.....	3
Table of Authorship.....	4
Table of Contents.....	6
Table of Figures.....	11
Table of Tables.....	15
1 Introduction.....	17
1.1 Background and Literature Review.....	18
1.1.1 CubeSat Applications and Missions.....	18
1.1.2 Review of CubeSats at WPI.....	19
1.1.3 Helmholtz Cage.....	21
1.2 Goals.....	21
1.3 Objectives, Approach, and Standards.....	22
1.4 Mission Operations and Constraints.....	25
1.5 Design Requirements and Constraints.....	27
1.5.1 CubeSat.....	27
1.5.2 Helmholtz Cage.....	37
1.6 Project Management and Budget.....	38
1.7 Task Breakdown and Timetable.....	39
1.8 Facilities and Equipment.....	40
2 Payload and Design Implications.....	41
2.1 Payload Overview.....	41
2.2 Payload Data Collection.....	44
3 Mechanical Design and Analysis.....	46
3.1 Mechanical Overview.....	46
3.2 Mechanical Requirements.....	46
3.3 Mechanical Design.....	48
3.3.1 Material Models.....	51
3.3.2 Summary of Mechanical Design.....	52

3.4 Mechanical Analysis	57
3.4.1 Analysis Setup	58
3.4.2 Modal Analysis.....	60
3.4.3 Random Vibration Analysis	61
3.4.4 Results	65
4 Propulsion Analysis	73
4.1 Propulsion Overview.....	73
4.2 Thruster Selection	75
4.3 Orbital Analysis.....	81
4.3.1 Science Leg Orbit	81
4.3.2 Mission Set-Up.....	83
4.3.3 Orbital Results	86
5 Power System Design and Analysis	92
5.1 Power Overview	92
5.2 Component Selection	95
5.3 Solar Array Power Generation Modeling.....	103
5.3.1 Solar Power Analysis.....	103
5.3.2 Solar Power Results.....	106
5.4 Dynamic Power Model.....	109
5.4.1 Dynamic Power Analysis	109
5.4.2 Dynamic Power Model Results	111
6 Telecommunications Design.....	116
6.1 Telecommunications Overview.....	116
6.2 Component Selection	117
6.3 Ground Station Selection	118
6.3.1 Recommended WPI Ground Station Set-Up.....	121
6.4 Access Time Calculation and Analysis.....	123
6.5 Results.....	125
7 Attitude Determination and Control Subsystem	130
7.1 Attitude Determination and Control System Overview	130

7.2 Attitude Determination and Control Component Selection	130
7.2.1 Magnetorquer Theory and Selection	132
7.2.2 Sensor and Onboard Computer Selection.....	135
7.3 Attitude Determination and Control Methods and Simulations.....	143
7.3.1 Detumble Methods and Simulations.....	143
7.3.2 TRIAD Method and Simulation	148
7.3.3 Proportional-Derivative and Trajectory Optimization Controllers	151
8 Space Environment and Effects	155
8.1 The Ionospheric Environment.....	155
8.1.1 Ionosphere Overview.....	155
8.1.2 Ionospheric Anomalies and Their Effects	159
8.1.3 Ionospheric Effects and Analysis	161
8.2 The Magnetic Field Environment	163
8.2.1 Analysis of the Earth’s Magnetic Field	164
8.2.2 Analysis of the Induced Magnetic Field by the Magnetorquers.....	165
8.3 The Radiation Environment	167
8.3.1 Radiation Analysis: Total Ionizing Dose.....	168
8.3.2 Radiation Analysis: Solar Energetic Particles	170
8.3.3 Radiation Analysis: Galactic Cosmic Rays	173
8.4 Particle Impacts	175
8.4.1 The Meteor Environment.....	175
8.4.2 The Debris Environment	177
9 Thermal Control Analysis.....	179
9.1 Thermal Overview.....	179
9.2 Thermal Analysis Methods	184
9.3 Results	194
10 Helmholtz Cage	204
10.1 Helmholtz Cage Overview	204
10.2 Helmholtz Cage Development and Design.....	205
10.3 Structural Design.....	211

10.3.1 Magnetorquer Mounting Method	215
10.3.2 Power Supplies	216
10.3.3 Motor Controllers	217
10.4 Magnetorquer Test Article	218
10.5 Data Collection Methods.....	219
10.6 Data Management	222
10.7 Assembly.....	224
10.8 Testing.....	231
11 Conclusion, Recommendations, and Social Implications.....	236
11.1 Conclusions	236
11.2 Recommendations	238
11.2.1 Spacecraft Charging	238
11.2.2 Integrated System	239
11.2.3 Magnetorquers and Spinner Satellites	239
11.2.4 Thermal analysis.....	239
11.2.5 Magnetorquer Test Article Stand	240
11.3 Social Implications of CubeSats	240
Bibliography	244
Appendices.....	260
Appendix A: ANSYS Custom Material Specifications	260
Appendix B: SpaceX Vibration Spectrum Deformation Results	261
Appendix C: MATLAB Script for Initial Estimate of ΔV	263
Appendix D: MATLAB Script for Determining ΔV Budget with Expanded Thruster Tank	264
Appendix E: STK Access Report Sample.....	265
Appendix F: ADCS Component Decision Matrices	268
Appendix G: B-dot Controller MATLAB Script	272
Appendix H: TRIAD Attitude Determination.....	274
Appendix I: Basic PD Attitude Controller	275
Appendix J: PD Attitude Controller.....	277
Appendix K: STK to COMSOL conversion MATLAB code.....	279

Appendix L: MATLAB Script for Plotting ADCS Helmholtz Cage Magnetic Field.....	281
Appendix M: MATLAB Script for Determining Viable Coil Configurations.....	282
Appendix N: Helmholtz Cage Instruction Manual	283
Frame Assembly:.....	283
Connection Set-Up:	289
Arduino	292
Sensor Connections, Wiring, and Code.....	293
Appendix 1: Arduino Code.....	308
Appendix 2: Python Code.....	315

Table of Figures

Figure 1: MagSTARS CubeSat.....	17
Figure 2: Rail-based dispenser vs tab-based dispenser [30]	28
Figure 3: Multiple CSDs attached to payload adapter [29]	29
Figure 4: Gantt chart	40
Figure 5: The mini-INMS flight and engineering units used for the Dellinger mission [44].....	41
Figure 6: Components for analyzing particle density within the mini-INMS [47].....	44
Figure 7: Maximum dimensions of the CubeSat (dimensions in mm)	49
Figure 8: Updated dimensions of CubeSat	50
Figure 9: Chaboche hardening parameters	52
Figure 10: Example 6U Tab CubeSat [30]	53
Figure 11: Exploded view of MagSTARS primary structural components.....	54
Figure 12: MagSTARS internal design.....	55
Figure 13: MagSTARS instrument field of view.....	56
Figure 14: MagSTARS inertia matrix.....	57
Figure 15: Meshed model of CubeSat	60
Figure 16: NASA GEVS random vibration spectrum [50].....	63
Figure 17: SpaceX random vibration spectrum [10].....	63
Figure 18: Comparison of vibration spectra	64
Figure 19: ANSYS block diagram.....	65
Figure 20: CubeSat-centric coordinate system [30].....	66
Figure 21: MagSTARS calculated vibrational modes	67
Figure 22: Stress in MagSTARS base plate.....	69
Figure 23: Y-axis GEVS deformation	70
Figure 24: X-axis GEVS deformation	71
Figure 25: Z-axis GEVS deformation.....	71
Figure 26: Initial state estimates for MagSTARS in Astrogator.....	84
Figure 27: MagSTARS' orbit paths.....	89
Figure 28: Apogee and perigee throughout mission.....	89
Figure 29: Graph of solar cell power output compared to solar cell current and voltage [56].....	93

Figure 30: ISISpace PDU system diagram [36].....	95
Figure 31: ISISpace MEPS system diagram.....	98
Figure 32: Fully assembled ISISpace MEPS stack.....	99
Figure 33: Four primary MEPS modules (PBP, PCU, PBU, PDU)	99
Figure 34: ISISpace 6U CubeSat platform	100
Figure 35: Solar cell – battery system overall efficiency at different input voltages [35].....	102
Figure 36: Model of MagSTARS in Blender.....	104
Figure 37: Solar cell illumination in STK [64].....	105
Figure 38: Plot of average power generation over mission for different orbits.....	106
Figure 39: Detailed power profiles before and after 120 days.....	108
Figure 40: Power use during constant propulsive maneuvers	112
Figure 41: Power use during periodic propulsive maneuvers.....	113
Figure 42: Power use near the end of the Science Leg	115
Figure 43: STK ground station coverage map	120
Figure 44: Indoor and outdoor unit diagrams [72].....	122
Figure 45: Software architecture [72]	123
Figure 46: NanoAvionics Magnetorquers MTQ3X full magnetorquer set employing two torque rods and one air core magnetorquer [72]	134
Figure 47: NewSpace Systems NCSS-SA05 [76]	136
Figure 48: Hyperion Technologies GNSS200 [77]	137
Figure 49: BEI Model 4311A linear accelerometer [78].....	139
Figure 50: Applied Technologies ARS-15 [77].....	139
Figure 51: ISIS On-Board Computer without daughterboard [78].....	141
Figure 52: Spacemanic LODESTONE SM-ADS-AG4 [79]	142
Figure 53: 6U payload initial tumble rates about each body axis [31]	146
Figure 54: Three detumble simulations from large angular rates	147
Figure 55: Basic PD controller attitude adjustment.....	152
Figure 56: PD controller failure to achieve desired attitude	154
Figure 57: Radio communication’s reliance on the ionosphere [87].....	156
Figure 58: Electron density of the ionosphere, highlighted F1 region [88].....	157

Figure 59: The Equatorial Ionization Anomaly [84]	160
Figure 60: Magnetic field intensity with respect to altitude for one science leg orbit of MagSTARS from STK	165
Figure 61: Magnetorquer and magnetic field modeled in COMSOL	166
Figure 62: Probability of SEP fluences on MagSTARS predicted by STK	172
Figure 63: GCR flux vs. particle energy level over MagSTARS' mission duration from STK.	174
Figure 64: Particulate meteor impact flux over MagSTARS' mission duration from STK	176
Figure 65: Absorbance and emittance of various materials [90]	181
Figure 66: Design of a louver, an example of active control [42]	182
Figure 67: Beta angle in relation to the Sun vector [90].....	183
Figure 68: MagSTARS STK model used for SEET thermal analysis	188
Figure 69: COMSOL model setup for internal dissipation.....	192
Figure 70: COMSOL model setup for external radiation	193
Figure 71: SEET temperature graph over 24 hours for the hot case.....	195
Figure 72: SEET temperature graph over 24 hours for the cold case.....	195
Figure 73: Sun facing exterior of the COMSOL hot case	196
Figure 74: Sun facing exterior of the COMSOL cold case.....	197
Figure 75: COMSOL internal components model for the hot case	198
Figure 76: COMSOL internal components model for the cold case	199
Figure 77: Side of MagSTARS opposite the Sun vector for the COMSOL hot case.....	200
Figure 78: Side of MagSTARS opposite the Sun vector for the COMSOL cold case.....	201
Figure 79: The rate of heating for the magnetorquer over 24 hours.....	201
Figure 80: Basic Helmholtz cage design	205
Figure 81: Magnetic field strength along the central axis of 1.44-meter coils	208
Figure 82: Number of cases that met the magnetic field requirements for a coil side length range of 0.825 to 0.875 m with a 0.025 m step size	211
Figure 83: Helmholtz cage corner attachment methods	213
Figure 84: Helmholtz cage bracket.....	214
Figure 85: Kungber DC power supply [108]	216
Figure 86: Cytron 30A, 5-30V Single Brushed DC Motor Controller	218

Figure 87: The Adafruit LIS3MDL Triple-Axis Magnetometer [110].....	220
Figure 88: The Adafruit LSM6DSOX + LIS3MDL [111]	220
Figure 89: Adafruit INA260 Sensor [112].....	221
Figure 90: Arduino MEGA 2560 [113]	222
Figure 91: Coil welding in progress.....	225
Figure 92: Coil corner following welding	226
Figure 93: Wire at a coil corner	227
Figure 94: Mounting bracket with rubber spacers	228
Figure 95: Coil banana plug connectors	229
Figure 96: Assembled Helmholtz cage.....	230
Figure 97: Helmholtz cage temperature reading.....	231
Figure 98: Helmholtz cage temperature plot	232
Figure 99: Helmholtz cage magnetic field test	233
Figure 100: Magnetorquer test article stand constructed from an air hockey table.....	235
Figure 101: NO and JOE.....	284
Figure 102: PAUL or SMOL connected to NO and JOE	285
Figure 103: SMOL and PAUL connected to NO and JOE.....	286
Figure 104: FRED or TED added to frame.....	287
Figure 105: Full frame assembly	288
Figure 106: Bracket and rubber pieces	289
Figure 107: Splitter	290
Figure 108: Wiring Diagram.....	291

Table of Tables

Table 1: Mission Leg Overview	25
Table 2: Relevant Specifications for the Mini-INMS Payload [5], [44].....	43
Table 3: Additional Component Placement Requirements.....	51
Table 4: AL7075-T6 Material Properties	52
Table 5: Center of Mass of MagSTARS.....	66
Table 6: NASA GEVS Vibratory Response	68
Table 7: SpaceX Vibratory Response.....	72
Table 8: Propulsion System Types [54].....	75
Table 9: First Down-Selection of Thruster Options and Their Parameters [22], [51], [52], [53].	78
Table 10: First Down-Selection of Thrusters' Required Propellant Mass and Total Impulse.....	79
Table 11: Lifetime of Various Potential Final Orbits of MagSTARS Without Stationkeeping...	82
Table 12: Maneuver Summary.....	87
Table 13: MagSTARS' Mission Timeline	88
Table 14: Power Subsystem Decision Matrix.....	97
Table 15: Subcomponent Power Requirements	110
Table 16: Radio Frequencies [2].....	118
Table 17: Ground Station Locations Provided by AWS.....	120
Table 18: Access Times by Location.....	124
Table 19: Access Time for Mission Duration.....	125
Table 20: Data Rate Calculation.....	126
Table 21: Transmission Statistics	127
Table 22: Receiving Statistics.....	128
Table 23: Sample Decision Matrix	132
Table 24: NanoAvionics Magnetorquers MTQ3X Specifications [73].....	135
Table 25: NewSpace Systems NCSS-SA05 Specifications [76]	136
Table 26: Hyperion Technologies GNSS200 [75].....	138
Table 27: Applied Technologies ARS-15 and BEI Model 4311A [78], [79].....	140
Table 28: ISIS On-Board Computer [80].....	141

Table 29: Spacemanic LODESTONE SM-ADS-AG4 [81].....	143
Table 30: Radiation Tolerances for All MagSTARS Components	169
Table 31: Radiation Dosage Over the MagSTARS Mission	170
Table 32: Mission Averages of Debris Impact Flux.....	178
Table 33: Thermal Limits of Selected Components	184
Table 34: Power Dissipation of Selected Components.....	187
Table 35: Hot Case Parameters.....	193
Table 36: Cold Case Parameters.....	194
Table 37: Simulated Temperature Comparison	202
Table 38: Sensor I2C Addresses.....	223

1 Introduction

In 1999, the CubeSat concept was developed by researchers at California Polytechnic Institute, San Luis Obispo, and Stanford University's Space Systems Development Lab. Their goal was to design a class of satellite that would provide university students hands-on access to the space industry [1], [2]. CubeSats are small, versatile satellites made up of 10 cm cubes, or "U's," and they are usually sent to orbit as secondary payloads on large launch vehicles [3]. Since 1999, over 1,300 CubeSats have been launched for a wide variety of scientific, educational, and commercial missions [4].

The goal of this Major Qualifying Project (MQP) is to develop the conceptual design of the Magnetospheric near Sun-synchronous Tumbling Attitude Research Satellite, or MagSTARS, as well as to design and build a Helmholtz cage to test a preliminary version of MagSTARS' magnetic attitude control system. MagSTARS, shown below in , is a 6U CubeSat designed to carry a miniaturized Ion Neutral Mass Spectrometer (mini-INMS) developed by NASA Goddard [5] in a near Sun-synchronous orbit to gather data on particle densities in the F layer of Earth's ionosphere.

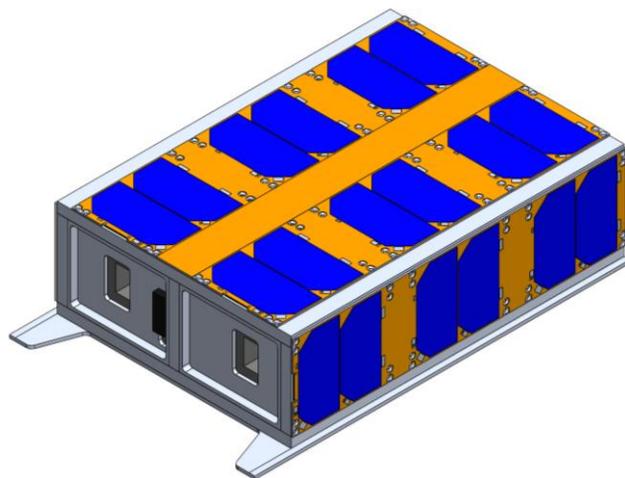


Figure 1: MagSTARS CubeSat

This MQP builds off the research done in several past CubeSat-focused MQPs, primarily the 2019 Systems Engineering Group's design of the NIMS eLEO Atmospheric Observer (NeAtO) [6], [7].

MagSTARS is assumed to be deployed from SpaceX's Falcon 9 launch vehicle at approximately 600 ± 25 km and an orbital inclination between 97.41 and 97.89 degrees [8]. It will then transfer to a near Sun-synchronous orbit with a 250 km perigee and 400 km apogee where it will record data on the ionosphere's F layer for approximately 120 days. Analysis of MagSTARS in this MQP consists of mechanical, propulsion, power, telecommunications, attitude determination and control (ADCS), environmental, and thermal analysis, as well as and physical testing of the ADCS subsystem.

1.1 Background and Literature Review

In this section, CubeSat applications, benefits, and past missions will be discussed in order to provide relevant context to this project.

1.1.1 CubeSat Applications and Missions

Since their creation in 1999, CubeSats have opened the door for many students, researchers, small companies, and even developing countries to take part in space exploration and research. Their manageable size and relatively low cost make them an ideal university research project, starter satellite, or technology demonstrator [9]. The cost to launch CubeSats is also significantly reduced because they are often deployed as secondary payloads on larger missions, and many companies offer dedicated rideshare programs which provide a convenient launch method for groups who would not otherwise have the opportunity [10].

Many educational and research-based CubeSat missions are developed through NASA's CubeSat Launch Initiative. The CubeSat Launch Initiative is a program which selects CubeSats designed by NASA centers, educational institutions, and non-profit organizations in the US for deployment from the International Space Station or as extra payloads on US Government launches. Since 2010, there have been 101 CubeSat missions flown under the CubeSat Launch Initiative with 220 missions selected [3]. These missions were from 102 different organizations across 41 states, which range from elementary schools and universities, to the Amateur Radio Association. These missions have conducted research on Earth's atmosphere, space weather, radiation testing, solar sails, biological sciences, and more [11].

In recent years, large commercial space entities have begun to take advantage of and promote the development of CubeSats. Several companies, such as SpaceX and NanoRacks, organize CubeSat deployments as secondary payloads on larger launches without the CubeSat having to be selected for NASA's program [9]. For example, Planet Labs has launched several "Flocks" of their Dove-series CubeSats on the Nano-Racks CubeSat deployer. These Earth-imaging satellites form several constellations over major agricultural or urban areas, providing valuable data for monitoring agricultural yields, tracking deforestation, analyzing urbanization patterns, and strengthening natural disaster relief [12].

1.1.2 Review of CubeSats at WPI

A series of CubeSat design MQPs have been completed at WPI, beginning with the Sphinx-NG CubeSat. This first MQP was developed as a cooperative project between WPI, NASA's Goddard Space Flight Center and the Space Research Center in Poland [13], [14], [15]. The objective of this project was to design a 3U CubeSat that would carry the Sphinx-NG X-ray

spectrometer and run solar/terrestrial X-ray spectrometry experiments once it reached its planned polar and Sun-synchronous orbit at an altitude of 600 ± 25 km. This CubeSat was planned with an 18-year lifespan in mind, allowing for a large quantity and variety of scientific data to be gathered, making best use of the satellite. Subsequent MQPs have, in 2013 [16], [17], [18] and 2017 [19], [20], [21], built on the initial concept laid out by the 2012 project. These MQPs continued the development of the Sphinx-NG CubeSat, finalizing component selection, running more extensive analysis for various subsystems, and making further design decisions based on what was learned.

Another project relevant to this MQP is the 2018 project that designed a CubeSat for an extreme low Earth orbit (eLEO) [22], [23], [24]. This 2018 MQP had two primary goals: the design and analysis for a 16U CubeSat capable of orbital maneuvers and formation flying at approximately 500 km altitude, and the design and analysis of a smaller 4U CubeSat flying in eLEO at approximately 210 km in a circular orbit, exploring a region where the solar wind energy couples to the Earth's atmosphere. Both CubeSats assumed an insertion at the International Space Station.

A third MQP of particular interest is the NeAtO CubeSat, developed in 2019. There are many similarities between NeAtO and MagSTARS, such as the chosen 6U configuration, the use of the mini-INMS instrument, and the overall science mission goal [6]. The target orbit for NeAtO was an elliptical orbit with an apogee of 440km and perigee of 200km, allowing the satellite to dip into the upper atmosphere obtaining science data from the lower ionosphere. Due to its low perigee, to meet the target satellite lifetime of 150 days, active drag compensation was required to increase the length of satellite operation. NeAtO was designed to fly with one 2U ram face always pointed in the direction of flight, with the instrument aperture mounted to said ram face. Because

NeAtO and MagSTARS missions are so similar, MagSTARS builds on work done by the NeAtO team, furthering the development of WPI CubeSat research.

1.1.3 Helmholtz Cage

A Helmholtz cage is a device that induces a magnetic field that simulates what a satellite would experience at different positions in orbit. The field is created via current running through conductive coils, as defined by the principles of the Biot-Savart Law [25], [26]. The Helmholtz cage is made up of several orthogonal square coils of wire. The field maintains an equal amount of power in all directions, due to the orthogonal placement. Square coils are used because they provide a more uniform field than circular coils [25]. The field needs to be controllable, as the strength of magnetic fields during a mission can vary significantly. The magnitude of the field is governed by the number of turns in the coils, the distance between coil pairs, and the magnitude of the current running through the wire [26]. The primary method of control is varying the current, as it is the easiest of these three variables to manipulate. A variable power supply is used to control the current and thus the magnetic field.

1.2 Goals

This MQP studies and advances the research of CubeSat systems at WPI by making improvements on research previously conducted by other CubeSat-based MQPs and furthering the technology available.

The primary goal of this MQP is to design the MagSTARS CubeSat for a magnetospheric science mission in a low Earth orbit (LEO). Specifically, this will entail designing a spinning CubeSat that makes use of the mini-INMS mass spectrometer, and flies in an elliptical near polar orbit with an initial apogee of 400 km and perigee of 250 km. The mini IMNS will collect data on

the particle composition of the F2 layer of the atmosphere in order to better understand its makeup. For the team to accomplish these goals, the spacecraft design was divided into the following subsystems: mechanical, propulsion, power, telecommunications, ADCS, the environmental/science payload, and the thermal control. For each of these subsystems, the focus was selecting and analyzing components that would lead to optimal satellite performance for the duration of the mission. While the work done on these subsystems relates to the design portion of the CubeSat, other issues related to the project such as the possible social and environmental impacts were also considered.

The secondary goal of this project was to design, fabricate, and demonstrate an Attitude Determination and Control System (ADCS) Test Cell, specifically a Helmholtz cage [25]. The purpose of this Helmholtz cage is to generate a dynamic magnetic field that replicates the conditions experienced by a satellite in Earth orbit, useful for testing magnetorquers. These components are reliant on the Earth's magnetic field for operation and create torques necessary for attitude control. This testing is necessary because Earth's magnetic field affects the hardware and software of the ADCS, so the Helmholtz cage allows for the verification of ADCS functionality. [27].

1.3 Objectives, Approach, and Standards

1. Develop mission operations
 - a. Mission operations will be developed to extended mission lifetime with optimal conditions for collecting and transmitting data.
 - b. Limiting factors from each subsystem will be considered and balanced to develop an optimal mission plan.
2. Develop MagSTARS design requirements

- a. The MagSTARS design shall follow requirements sourced from the SpaceX User Manual and the Planetary Systems Corporation Payload Specification.
3. Perform mechanical design
 - a. Develop a CAD model of the CubeSat using SolidWorks; the design shall be maintained for usage by other subsystems.
 - b. The CubeSat shall conform to all required vibration specifications.
4. Perform payload integration
 - a. The payload shall correctly interface with the spacecraft and will be operational when desired under mission conditions.
5. Perform trade study on propulsion systems
 - a. The trade study shall assess the parameters of propulsion systems to determine the best choice for MagSTARS' requirements and constraints.
 - b. ΔV , propellant mass, and thrust budgets for the mission shall be determined.
6. Perform orbital analysis
 - a. Orbital analysis shall be conducted using the Satellite Tool Kit (STK) to optimize lifetime in the science phase of mission by determining and analyzing a final orbit and implementing orbital stationkeeping.
 - b. Orbital analysis shall determine the ΔV , propellant mass, and burn times required for the mission.
7. Analyze power system characteristics
 - a. The power analysis shall create a comprehensive generalized power budget for all subsystem components to be used for initial power system selection.
 - b. The power analysis shall model solar panels using STK to assess solar array power collection in various orbits.
 - c. The power analysis shall ultimately create a dynamic power model for all satellite systems using MATLAB.
8. Select MagSTARS' power system
 - a. Component selection shall entail utilizing a decision matrix comparing various power systems based on their stated performance characteristics.

- b. Power components shall be selected to support satellite functionality based on power budget and orbital analysis.
- 9. Analyze thermal characteristics of the mission
 - a. Thermal limits of onboard components shall be determined and verified for compliance using COMSOL and STK.
 - b. Radiation sources acting on MagSTARS during flight shall be modeled and analyzed using COMSOL and STK.
 - c. Appropriate means of thermal control shall be selected.
- 10. Perform environmental effects analysis
 - a. Potential environmental hazards to the mission and common ways to mitigate their effects shall be identified.
 - b. Analysis shall be performed on models of environmental factors through COMSOL and STK to understand their effects on the mission.
- 11. Perform telecommunications design and analysis
 - a. The hardware components shall be selected to produce an optimal link budget.
 - b. The design shall optimize the access time of the satellite via ground station locations.
 - c. Analysis of access times will be performed using STK
 - d. The design shall detail requirements for an operational ground station.
- 12. Develop attitude determination and control system
 - a. Components necessary for attitude determination and control actuation shall be selected.
 - b. Control schemes for attitude determination and control shall be developed in MATLAB and STK.
 - i. Detumble
 - ii. Spin state attitude control
 - iii. Attitude determination method
 - c. Simulations of control schemes shall be performed in MATLAB and STK.
- 13. Design and build a Helmholtz cage
 - a. A working Helmholtz cage shall be designed using SolidWorks and constructed to simulate Earth's magnetic field during the MagSTARS mission.

- b. The Helmholtz cage shall be designed so it is simple to reconstruct for easy storage.
- c. Data on the response of a magnetorquer test article to the cage’s simulated magnetic field shall be collected.

1.4 Mission Operations and Constraints

MagSTARS’ mission is broken down into three major legs in order to better describe the operations required throughout the mission. Legs are chosen based on changes in orbit, and each leg can be further broken down into phases.

Table 1: Mission Leg Overview

Mission Leg	Phase	Operations
Ejection	-	Ejection from Falcon 9, detumbling
Transfer	-	Transfer orbit injection burn, transfer orbit, science orbit insertion burn
Science	Static	Not spinning, attitude adjustments conducted during this phase
	Spin-up	Increasing angular velocity to desired spin rate
	Spinning	Spinning, science data collected during this period
	Spin-down	Decreasing angular velocity to return to static phase

The Ejection Leg is outlined as beginning with deployment from the Falcon 9 launch vehicle and ending right before MagSTARS’ transfer orbit injection burn. During this leg, the CubeSat will be ejected from the Falcon 9 where it enters the initial orbit. Most of the onboard equipment will come online and the telecommunications subsystem will establish a connection

with the nearest ground station and began relaying attitude and position data. The ADCS will shed any residual angular velocity from the deployment and correct MagSTARS' attitude using the magnetorquers in a process known as detumbling. No science data will be collected during this period.

Once MagSTARS stabilizes and is ready for its first burn, the Transfer Leg begins. This burn puts MagSTARS on a transfer orbit from its initial 500 km circular orbit to its final 250 km perigee by 400 km apogee orbit. This leg includes the entire transfer orbit period. The Transfer Leg takes place while MagSTARS is fully exposed to the Sun, so the propulsion system will be powered by the solar arrays. MagSTARS will maintain communication with the ground throughout this leg, though no science data will be collected or transmitted. The Transfer Leg ends with the transfer's second burn, during which MagSTARS enters its 250-400 km orbit.

The Science Leg covers the entire science period of the mission. This leg is broken down into different operating phases: static, spin-up, spin, and spin-down. During the static phase, MagSTARS is in its final orbit but not spinning. Attitude adjustments will be made during this phase, and data will be transmitted as well, but science data will not be collected. The spin-up period occurs during the transition from the static phase to the spin phase and involves using the magnetorquers to gradually increase MagSTARS' angular velocity until the desired spin rate is achieved. Once spin-up is complete, the spin phase begins and the mini-INMS payload will turn on and begin collecting data. Because of the location of the antenna, it was determined that science data can be transmitted during this phase as well, as the spinning motion will not affect the transmission. The majority of the mission duration is spent in this operating phase. The final operating phase is spin-down, or the transition from the spin phase to the static phase. The payload will be shut off during this period, and the magnetorquers will be used to slow the rotation rate

until the static phase is entered again. MagSTARS will cycle through these four operating phases during the Science Leg, entering the static phase for any required attitude adjustment and then returning to the spin phase for the remaining mission duration.

1.5 Design Requirements and Constraints

This section presents two sets of requirements and constraints: one for the design of the CubeSat and mission, and one for the Helmholtz cage.

1.5.1 CubeSat

The following constraints, grouped by subsystem, apply to the MagSTARS CubeSat mission and were used to drive the design and development process.

1.5.1.1 CubeSat Dispenser Background and Selection

CubeSats are not directly attached to a launch vehicle, but rather reside within dispensers until the desired orbit is reached. Then, the dispenser opens and ejects the CubeSat, imparting an initial velocity, as well as some tumbling motion. As CubeSat missions have become more common, more types of dispensers have been developed. For 6U CubeSats, two main dispenser designs exist. The first is a four-rail system, similar to those used for 1 to 3-unit CubeSats, where the satellite must have a rail at each edge. These rails fit inside the dispenser, and the satellite slides out when deployed [28].

The second type of 6-unit dispenser uses a two-tab system, in which the satellite has two thin tabs running the length of the satellite which are clamped by the dispenser until release. Tab-based dispensers do not allow for multiple independent payloads to be placed in parallel within the dispenser, as contact to each tab is required. They do, however, allow for multiple payloads to

be placed in sequential order, so long as each is at least 50 mm long [29], [30]. Tab-based systems apply a preload to the tabs that run the entire length of the payload, clamping it in place. This ensures the payload cannot shift or vibrate significantly inside the dispenser and simplifies modeling for random vibration testing [30]. Overall, a tab-based dispenser was selected because it would more securely constrain the payload, as seen in Figure 2.

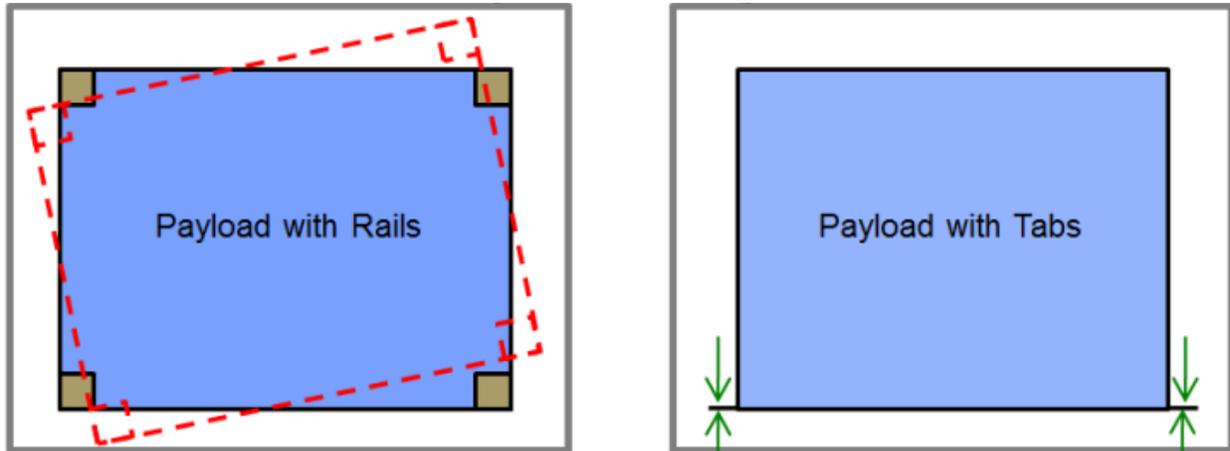


Figure 2: Rail-based dispenser vs tab-based dispenser [30]

Planetary Systems Corporation (PSC) pioneered the tab-based CubeSat specification which led to their development of the Canisterized Satellite Dispenser (CSD) [30]. Along with the previous justification for choosing a tab-based dispenser, these factors led the team to choose the CSD for this project. In addition, several documents are available comprehensively describing the CSD [29], [30]. Of special interest were practical tests done by PSC, in which a payload was deployed within a zero-G plane [31]. From this experiment, data on initial rotation rates were collected. This provided an empirical backing to the assumed initial conditions at deployment from the PSD.

1.5.1.2 Launch Vehicle Background and Selection

While launch vehicle selection was not a major focus of this project, a realistic launch vehicle was selected in order to determine an appropriate starting orbit altitude and inclination. The CSD has flown on several launch vehicles, including Electron, Atlas 5, Falcon 9, PSLV, and SPARK* [32]. Due to the availability of information and the ability to launch directly to a Sun-synchronous orbit, a SpaceX Falcon 9 rocket was chosen. MagSTARS is assumed to launch on a dedicated rideshare flight and be attached to a mechanical interface ring. This could be either a direct connection, or as one of many CubeSats, as shown in Figure 3 [29].

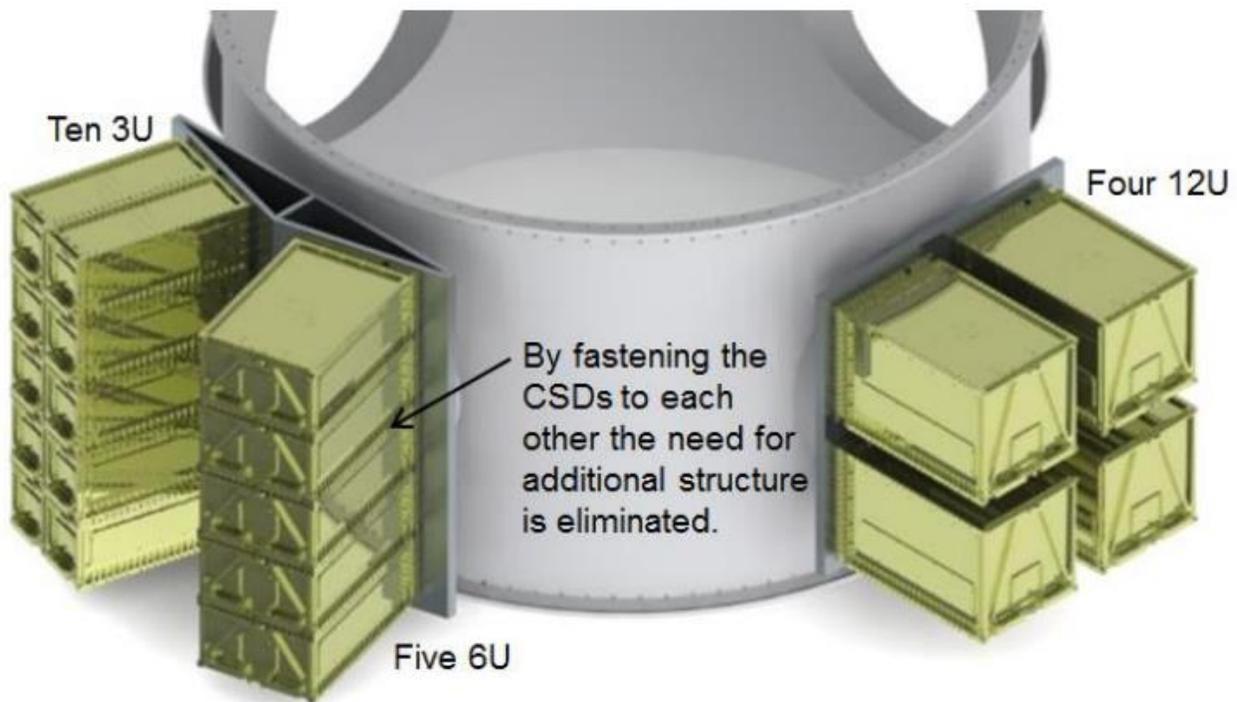


Figure 3: Multiple CSDs attached to payload adapter [29]

A Sun-synchronous orbit is a specific kind of polar orbit which precesses with the same period as the planet's solar orbit. This means that the spacecraft reaches perigee at approximately the same local time on each orbit [33]. This is desirable, as the same face of MagSTARS will

theoretically always point toward the Sun. This provides an almost constant power flow throughout the mission and simplifies power management. According to the Falcon 9 rideshare website, the initial orbit will be Sun-synchronous to a precision of 1 degree and have both an apogee and perigee between 500 and 600 kilometers with a precision of 25 km [8].

However, this does not consider orbital drag due to the atmosphere, which results in a gradually changing orbit. From the initial orbit, MagSTARS transfers to the science orbit, which has an apogee of 400 km and a perigee of 250 km. The apogee and perigee of this orbit gradually decrease over time, and the orbit is no longer non-Sun-synchronous. The CubeSat remains in full Sunlight for approximately 120 days, but after that point, it experiences shade for part of the orbit. This is discussed further in Section 5.

1.5.1.3 Mechanical

The mechanical subsystem objectives are to ensure that MagSTARS conforms to numerous mechanical requirements defined by the dispenser, the CubeSat specification, and the launch vehicle. These mechanical requirements were broken down into three subsections: Mechanical features, envelope and general properties, and vibration response.

Mechanical features include the tabs, which have several requirements for dimensions and materials defined by the CSD datasheet [29]. To maintain a proper load path to secure the CubeSat, it is critical that the tabs were the correct size, thickness, smoothness, and strength for them to be clamped by the CSD [30]. Also included in this category are contact points for the back of the dispenser, which must constrain the center of mass and avoid holes in the back wall. Finally, this category includes requirements for other necessary features such as removable electrical inhibitors.

Envelope and general properties requirements include a maximum envelope the CubeSat must always remain within, even when under vibrational load. This category also includes requirements for location of center of mass, venting volume per area, debris, and collectable condensed volatiles. Notably, there is no maximum mass requirement. Mass is instead constrained by the peak tab loading, discussed in the next paragraph, which is additionally highly dependent on damping and payload construction [30].

Finally, vibrational requirements include several tests that must be passed. The first of these is peak tab loading. The maximum load on the tabs, at 3 standard deviations, must not exceed 3559 N. This requirement included both random vibration and launch loads, in all primary axes [30]. Another vibrational requirement that was met is maintaining structural integrity when being subjected to various random vibration spectra. Requirements for this property are defined in the CSD payload specification, as well as SpaceX's User Manual. Each of these sources had slightly different required Power Spectral Density (PSD) curves, so MagSTARS was required to conform to both requirements. The final vibrational requirement was modal analysis, which determines the primary vibrational modes of MagSTARS. These modes must not be below 40 Hz, as per the SpaceX Payload User manual [10].

Some of these requirements, such as tab design, are independently defined and the CubeSat could be designed to meet them from the beginning. These requirements informed how the spacecraft model was created. Other requirements, such as center of mass envelope and peak tab loading, could not be verified before a full-featured CAD model of the CubeSat was created and analyzed. For this project, ANSYS simulation software was used to create finite element models of the spacecraft, conduct simulations of the expected loading environment, and produce results,

including tables of the primary vibrational modes, 3D models of deformation, and numerical results for tab loading.

An iterative process was employed for the mechanical design process. As the design progressed, more high-fidelity models were produced and tested. Results from these tests were used to inform design decisions for upcoming models. Center of mass and vibrational response were evaluated, and changes made to the model to address issues.

1.5.1.4 Propulsion

For MagSTARS, there are constraints on the amount of power used by the chosen propulsion system and on the mass and volume of the propellant—the latter constraint being necessary due to the small size of the CubeSat. MagSTARS is planned to be deployed in a Sun-Synchronous Orbit (SSO) at an altitude of 500-600±25 km by the SpaceX Falcon 9 [8]. From there, it will enter a transfer orbit to bring it to its final orbit with a 400 km apogee altitude and 250 km perigee altitude, where it will stay for the remainder of its science mission. Enough propellant is needed in order to insert MagSTARS into its final orbit and maintain that orbit. To perform this insertion, the thrusters need to provide two impulses: one that takes MagSTARS from its initial orbit into its transfer orbit and another that takes MagSTARS from its transfer orbit onto its final orbit. Since a main goal of this mission is to remain in this orbit as long as possible to optimize the Science Leg lifetime, propulsion is also needed to maintain the orbit and make needed corrections during the mission, mostly to compensate for drag due to the low altitude.

Using MATLAB and the initial and final orbit altitudes and inclinations (described in Section 4.1), the required ΔV budget, or the total change in velocity during maneuvers, to achieve MagSTARS' Science Leg orbit was determined. Analysis using Systems Tool Kit (STK)

confirmed this budget and estimated the time it took to reach the final orbit and the amount of propellant needed to perform this maneuver. Once the assessment was completed, the initial propellant mass value was used to determine the amount of remaining propellant. Then, STK was used to further analyze and estimate how long that amount of remaining propellant could keep MagSTARS in its desired orbit. STK also allowed for inputs such as dry mass, fuel tank inputs, and engine model type, so propellant mass and volume constraints previously mentioned were considered in STK. This made it the main tool used for analyzing and optimizing the CubeSat and propulsion parameters in order to prolong the Science Leg of the mission for as long as possible.

The power constraint, as discussed in Section 1.5.1.5, had to be considered as well, since it impacted the decision of overall propulsion system type. Factors such as atmospheric drag also were considered, as enough thrust needed to be provided in order to compensate for the drag throughout the mission. MagSTARS' spinning motion was also a factor in the amount of drag experienced, which varied based off which of its axes was chosen for it to spin around.

1.5.1.5 Power

The power subsystem's objective is the successful collection, storage, management, and distribution of electrical power to support optimal functioning of all satellite subsystems while completing the science mission. This entails the selection of physical hardware, including solar panels, batteries, and power management electronics, that can fulfill the various power requirements of all subsystems, as well as an analysis of MagSTARS' power collection and use throughout its operation [6]. Due to the nature of this subsystem, and the changing power requirements of an evolving satellite design, component selection and power analysis was iterative, ensuring the needs of the current design are best fulfilled by the power subsystem.

The power subsystem has three primary constraining factors: power generation, storage, and use. For power collection, the near Sun-synchronous orbit selected for the mission kept the satellite in constant Sunlight during the first 120 days of the Science Leg, allowing the solar cells to provide continuous power to all subsystems during this period. However, due to the rotation of MagSTARS, only body mounted solar panels could be used, limiting the power collection of the panels. As the axis of rotation of MagSTARS was pointed almost directly at the Sun, it was expected that one 6U face would be fully exposed to Sunlight for the first 120 days of the mission. Based on available premade 6U solar panels, a maximum of 15–19 W of power is expected to be supplied during rotation. For power storage, batteries are heavy and were limited to only the necessary amount of power storage for the mission. This weight limitation is due to the cost of launching things to orbit, as well as the increased fuel needed during orbital maneuvers with additional weight [8]. A 30–40 Watt-hour battery based on commercially available battery systems was used, putting a hard limit on power storage [34], [35], [36]. Finally, the power use of subsystems throughout the flight governed power availability to other subsystems during the mission as certain systems power up or standby, causing power usage to vary dynamically. All these constraining factors were analyzed by comparing projected power generation simulated in STK to projected power use based on a detailed budget for subsystem power requirements. This analysis was used to determine the optimal power system for this mission by comparing component choice with projected performance.

1.5.1.6 Telecommunications

For this project, the telecommunications system design requirements included a link budget analysis, ground station determination, and hardware component selection. For these components

and analyses to be determined, a few requirements and constraints needed to be considered. All communication between satellites and ground stations needed to occur over a radio frequency (RF) per federal law with appropriate licensure [37]. This limits the components to transmitting and receiving data over RFs, such as Ultra-High Frequency (UHF), Very-High Frequency (VHF) and S-Band. Coverage and access time are requirements that limit the ability to transmit data. Ground stations are responsible for the “access time” which refers to the timespan that the ground station can communicate with the satellite. This access period allows for the transmitting of data such as instructions, telemetry, or science data. As a result, ground station location, relative to the satellite’s orbit, is crucial to maximize access time. The mission tactics, such as satellite command uploads or payload data downloads, would then need to be planned around this the access time to use the time effectively [38]. The components selected for this subsystem were required to comply with the thermal and power constraints of the CubeSat, as described in Sections 1.5.1.5 and 1.5.1.9, and had to be a reasonable size for due to the structural constraints, described in Section 1.5.1.3.

1.5.1.7 Attitude Determination and Control System

The Attitude Determination and Control System (ADCS) was responsible for determining and correcting the orientation of MagSTARS for each phase of its mission. These phases consist of deployment and detumbling, orbital maintenance, and scientific spinning motion. In order to meet the scientific and mission critical requirements for each of these phases, an ADCS system must be able to produce various desired attitude states while complying with system requirements set by other subsystems. All components of the ADCS must comply with the mass and volume requirements of a 6U CubeSat. Additionally, all components must comply with thermal conditions and power budgets set by the analysis performed for each of those subsystems.

The ACDS consists of two subsystems. The first of these is a suite of sensors which is utilized for attitude determination. The second, the control subsystem, uses actuators to apply torques that orient the spacecraft to the desired attitude. In connection with the requirements for the Helmholtz cage, it was decided that the only control actuators on MagSTARS would be magnetorquers. By combining these two subsystems with a control scheme, the attitude of MagSTARS can be fully controlled despite influences from sensor and actuator error, as well as external forces.

1.5.1.8 Environmental Effects

The environment of space is very harsh and creates many risks that must be accounted for to ensure a successful satellite mission [39]. Because of MagSTARS' low orbit, atmospheric drag plays a major factor in orbit maintenance and must be modeled to ensure the satellite's orbit does not decay before its ideal end of life [40]. MagSTARS relies heavily on magnetorquers, which generate a magnetic field to interact with Earth's and orient the satellite. The magnetometers onboard MagSTARS could be adversely affected by the magnetic field generated by the magnetorquers, therefore an analysis of the effects on the satellite must be completed to determine if magnetic shielding is required [6]. MagSTARS will be exposed to particle radiation while in the ionosphere, therefore the components' radiation tolerance should be compared to estimates of MagSTARS' lifetime radiation exposure to ensure electronics and other sensitive instruments will not be damaged from irradiation or ionization during the mission. [41]. Thermal radiation, another major environmental factor, is addressed in the next section.

1.5.1.9 Thermal Control

The thermal control subsystem was designed to ensure the CubeSat remains within its operational temperature range, despite the extreme temperatures of space. Temperatures can fluctuate rapidly and in large amounts in a vacuum and thus, there must be care taken to ensure that these temperatures do not exceed the thermal limits of the spacecraft [42]. Each component onboard MagSTARS has an operational temperature that must be maintained during its intervals of operation or the device would experience a reduction in functionality. Each component also has a survivability temperature that must be maintained throughout the duration of the mission. If this range is exceeded there would be permanent damage to the device. To ensure the temperature of MagSTARS stays within this range during its flight, several steps must be taken. First, the magnitude of the heat irradiated from the environment onto MagSTARS must be analyzed by modeling its orbit. Then, the internal heat generated by devices onboard MagSTARS must be analyzed and modeled. After these analyses, the parts of the spacecraft that exceeded their desired temperature ranges must be identified. Finally, the most effective and efficient methods of thermal control must be determined to maintain an acceptable temperature across the entirety of MagSTARS.

1.5.2 Helmholtz Cage

The Helmholtz cage is used to simulate the ambient magnetic field conditions that a satellite or satellite component will encounter during its mission. As such, the Helmholtz cage was required to produce a magnetic field equal to the maximum field strength MagSTARS would encounter, or approximately that of the Earth's magnetic field at altitudes of interest to the mission. Additionally, the Helmholtz cage had to be able to produce such a magnetic field over a volume

that could fully contain a 6U CubeSat or any of its individual components. This set the minimum volume as a cube with side length 0.3 m. To simulate a changing magnetic field throughout an orbit, the Helmholtz cage was controlled through a computer system.

To validate tests done in the Helmholtz cage, a magnetometer was used to measure the magnetic field produced by the cage. This measurement was used to calibrate the cage and ensure that true magnetic field conditions were as close to target conditions as possible. Additionally, the Helmholtz cage was designed such that it was able to be easily disassembled and stored with the coils parallel to each other. This disassembly does not include any complex tasks such as unwinding the coils or the use of any tools.

1.6 Project Management and Budget

This MQP team was separated into the following subsystems for the CubeSat portion of the project:

- Mechanical
- Propulsion
- Power
- Telecommunications
- Attitude Determination and Control
- Environmental Effects and Payload
- Thermal Control

For the ADCS Helmholtz cage portion of the project, this MQP team was split up into two groups:

- Design and Instrumentation
- Building and Data Handling

The project group met multiple times per week to discuss factors related to the entire CubeSat system and the Helmholtz cage. To keep the team on-track, a Project Manager role was held. Every three weeks, the responsibilities of this role were passed on to a different member of the team. The Project Manager was responsible for leading the team and taking meeting minutes during meetings and ensuring everything was in-line with the Gantt Chart in Section 1.7.

Tasks that needed to be performed by one subsystem, requested from a different subsystem, were kept track of in an Excel sheet listing different action items and their open and close dates. During weekly meetings with the project advisor, each subsystem lead presented their progress from the past week and discussed what they planned to do by the next weekly meeting.

The budget for this project, used for the Helmholtz cage portion, was \$250 per team member (\$1,750 total).

1.7 Task Breakdown and Timetable

The Gantt Chart in Figure 4 shows the tasks the team completed for this MQP project and the timeline for each task. As mentioned in Section 1.6, the Project Manager was responsible for keeping track of these tasks.

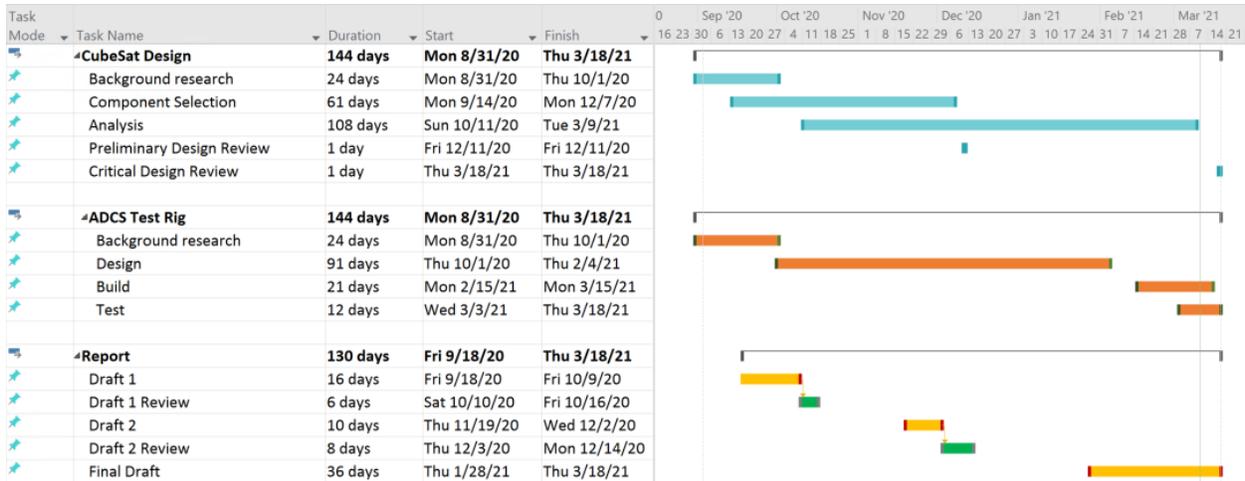


Figure 4: Gantt chart

1.8 Facilities and Equipment

The Helmholtz cage was constructed using the funds, equipment and facilities provided by the Aerospace Engineering Department. The cage will be housed in Higgins Labs 016. Due to COVID-19 concerns during the construction, team members were equipped with personal protection equipment (PPE) including face masks, face shields and disinfectant. WPI owned software subscriptions such as the Systems Tool Kit, COMSOL and MATLAB were also used for analysis. Additionally, the videotelephony application Zoom was used extensively over the course of this project.

2 Payload and Design Implications

To achieve the science goal of analyzing particle densities in the F layer of the magnetosphere, MagSTARS was designed to carry the miniaturized Ion Neutral Mass Spectrometer (mini-INMS).

2.1 Payload Overview

The mini-INMS was developed by NASA Goddard Space Flight Center with the goal of creating a small mass spectrometer that could effectively gather atmospheric particle density data on a CubeSat mission. The mini-INMS, pictured in Figure 5, uses time-of-flight mass spectrometry to measure the relative and total densities of various ion and neutral species in the atmosphere. Time-of-flight mass spectrometry is a scientific process in which the mass of a particle is measured using its kinetic energy, and the time at which the mass was measured is recorded. This made the mini-INMS the ideal instrument for this mission, as the particle data coupled with MagSTARS' location at the recorded time-of-flight will give insight into particle density data in different areas of the ionosphere [43].

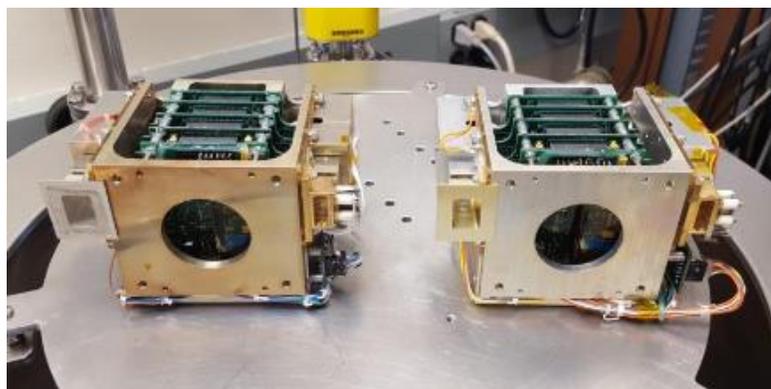


Figure 5: The mini-INMS flight and engineering units used for the Dellingr mission [44]

Several versions of the mini-INMS have been flown on CubeSat missions since its development. The first was ExoCube 1, a 3U weather satellite designed by California Polytechnic State University and launched in 2015. The launch was successful, but a failure in antenna deployment caused issues with data transfer. However, in-situ ion measurements were still performed, meaning the payload operated as planned [44]. A re-designed ExoCube 2 is set to launch soon with the goal of completing the original science mission [45]. The mini-INMS was also launched on the 6U Dellinger CubeSat in 2017. This mission was plagued with issues from the beginning, including malfunctioning Sun sensors, loss of GPS function, and uncontrolled spinning. After a year of troubleshooting, the Dellinger team was able to correct the issues and gather ion data from the mini-INMS, although the payload's neutral particle mode remained inoperative [46]. The most recent version of the mini-INMS will be flown on NASA Goddard's petitSat in 2021. The goal of this mission is to study irregularities in the mid to low altitude ionosphere, which will be achieved by a variety of science instruments including the mini-INMS [5]. The mini-INMS was improved for petitSat, and this is the version that will be used on MagSTARS. Relevant specifications of the mini-INMS are shown in Table 2.

Table 2: Relevant Specifications for the Mini-INMS Payload [5], [44]

Parameter	Performance
Field of View (effective)	± 10 deg around ram face
Volume	1.3U
Dimensions	8.3 cm x 9.2 cm x 16 cm
Mass	960 g
Power	1.8 W
Nominal Data Rate	13.7 kbps
Electrical Interface	± 5 V, ± 3 V, 12 C and SPI serial communication
Ion Species	H ⁺ , He ⁺ , N ⁺ , O ⁺ , NO ⁺ , O ₂ ⁺
Ion Range	10^3 cm ⁻³ to 10^9 cm ⁻³
Neutral Species	H, He, N, O, N ₂ , O ₂
Neutral Range	10^5 cm ⁻³ to 10^9 cm ⁻³
Sampling Time Rate	0.1s-10s
Default Sampling Rate	1s
Operating Temperature	-10 °C to 50 °C

The petitSat mini-INMS has two apertures on its ram-facing side for collecting particles with a 10° field of view. Both apertures open into chambers that can operate in either neutral or ion mode. In ion mode, pre-acceleration by a specific voltage gives all the ions the same energy as they enter the aperture. The velocity of each ion is measured by the time of flight over the distance they travel, and this known velocity and energy can be used to solve for the mass of each ion using kinetic energy. The calculated mass of each ion can be used to identify its species. In neutral mode,

a potential grid blocks ions from entering the aperture. Neutral particles are collected and ionized with an electron beam as they pass through the aperture. The neutral particles are then identified using the same process as the ions [47]. Figure 6 shows the layout of components within the mini-INMS.

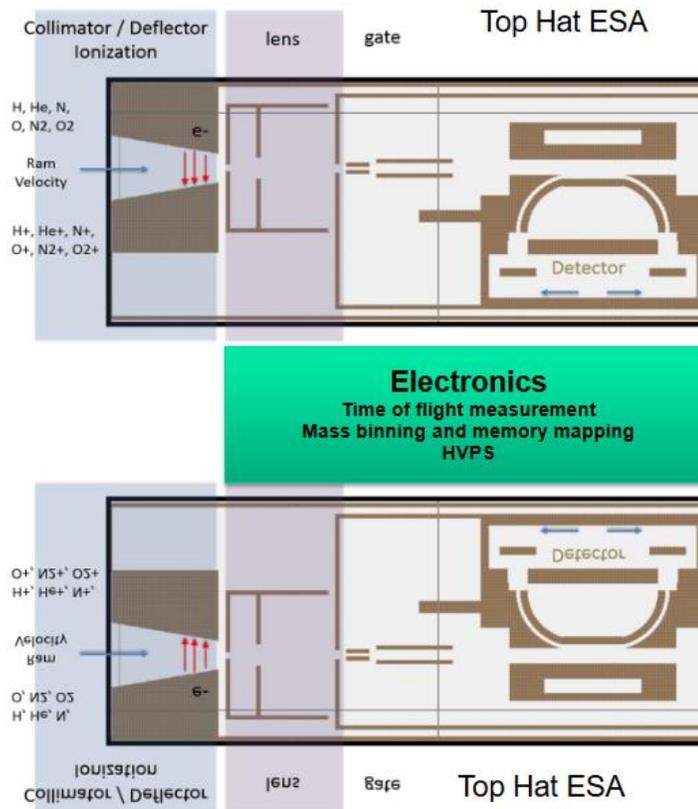


Figure 6: Components for analyzing particle density within the mini-INMS [47]

2.2 Payload Data Collection

MagSTARS will orbit in the F layer of the ionosphere, the highest ionosphere layer characterized by levels of ionization that shift with the time of day, seasons, and solar activity. The mini-INMS will collect the ionized particles of the F layer as it orbits and record the different

species of particle collected. The spinning motion of MagSTARS will act as a filter to collect only ionized particles, as the neutral particles cannot enter the aperture due to the spinning motion.

The F layer was chosen as the orbital altitude for this mission because of its importance in radio communication and the lack of empirical data available to fully model this area of the atmosphere. Radio signals are bounced off the plasma in the F layer to reach further distances around Earth's curvature, and perturbations or anomalies in this plasma can negatively impact these signals. The F layer is far from uniform, and all its variations have not been documented. The MagSTARS mission will gather valuable data on these variations and will supplement existing data on the full composition of the F layer. Further research on the ionosphere, the F layer, its anomalies, and the justification of this mission can be found in Section 8.1.

3 Mechanical Design and Analysis

This chapter discusses the mechanical subsystem of MagSTARS, and the analysis undertaken to verify the performance of this subsystem.

3.1 Mechanical Overview

The mechanical and structural analysis component of the project focuses on ensuring that the CubeSat meets all mechanical and structural requirements defined by the dispenser, the CubeSat specifications, the launch vehicle, and NASA. Some of these requirements are design-centric, such as center of mass restrictions, and can be easily checked for compliance. Other requirements, such as random vibration response, require further analysis. These requirements were tested with structural analysis simulations conducted using ANSYS software. Following assumptions made in previous CubeSat projects, it was assumed that individual off-the-shelf components had already been tested and conform to the stated requirements [19]. Thus, these components were modeled as rigid bodies, allowing for simpler and more accurate analysis of the structural framework, the primary focus of this subsystem.

3.2 Mechanical Requirements

The mechanical requirements were broken down into three subsections: mechanical features, envelope and general properties, and vibration response. Many of these requirements are listed in several places, including the CSD payload specification, the CubeSat specification, and the SpaceX user manual. The most stringent requirements are listed below:

Mechanical Features

- Tabs are required to run the whole length of the payload and have no holes, with some exceptions. If tabs are reduced according to these exceptions, allowable loading decreases proportionally [30].
- Tabs must be made of aluminum alloy with yield strength greater or equal to 56ksi, be hard anodized according to MIL-A-8625, Type III, Class 1, and must also have a max surface roughness of N7 [30].
- Tabs must meet dimensions and tolerances described in the CSD payload specification [30].
- The -Z face must either be solid or have discrete contact points enclosing the center of Mass, as this face makes contact with the dispenser [30].
- Any deployables must be verified with the CSD before flight [30].

Envelope and general properties

- Within the CSD payload specification, a maximum dynamic envelope is described. The payload must always remain inside this envelope, including when under thermal or mechanical load [30].
- The center of mass must be within a defined rectangular envelope in the center of the envelope [30].
- No maximum mass is defined. Instead, the limiting factor is tab loading, described in the previous section [30].
- During ascent, ventable volume per area must be less than 2000in [28].
- CSD de-pressurization rate will be 1.0 psi/s at maximum. The payload must be able to tolerate this [30].

- No debris may be generated [28].
- Total Mass Loss must be less than 1% [28].
- Collected Volatile Condensable Material must be less than 0.1% [28].

Vibration response:

- Modal analysis was performed to determine dominant modes [30]. The Payload axial and bending modes should be greater than 40hz [10].
- The payload was subjected to random vibration along each of its three primary axes. It met several requirements defined in NASA's General Environmental Verification Standards (GEVS) [30]. It also met requirements defined by SpaceX's user manual [10]. See Figure 18 for graphs showing these response requirements.
- The tab loading was within specification: loading was less than 3559N when subjected to either random vibration spectrum [30].

3.3 Mechanical Design

To meet the requirements discussed above in Section 3.2 a payload structure was designed, to which all other components were attached. This structure interfaced with the dispenser using the tab method. Beyond tab specifications, the CSD payload specification provided a maximum dynamic envelope for the structure. A cross section of this envelope is shown below in Figure 7.

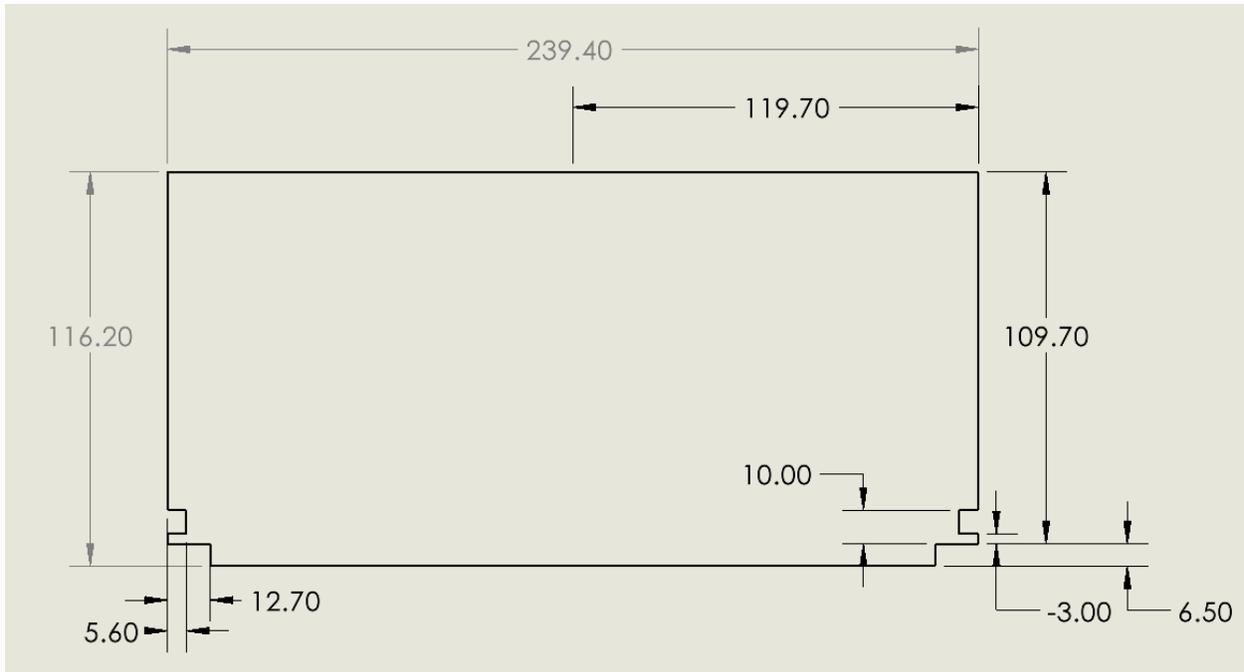


Figure 7: Maximum dimensions of the CubeSat (dimensions in mm)

This cross section provides the maximum internal space allowed by the CSD specification. However, due to its complex design, it was decided to simplify this cross-section. The only required elements are the tabs themselves, the rest of the shape can be changed as needed. The extra space at the bottom was judged as unnecessary, and thus the bottom was changed to be a flat plate. Similarly, the extra space at the sides was also deemed unnecessary and difficult to utilize effectively, and the payload was made less wide. These changes resulted in a payload with less volume, however, it was a simpler model and would result in easier construction. The updated cross-section is shown below in Figure 8.

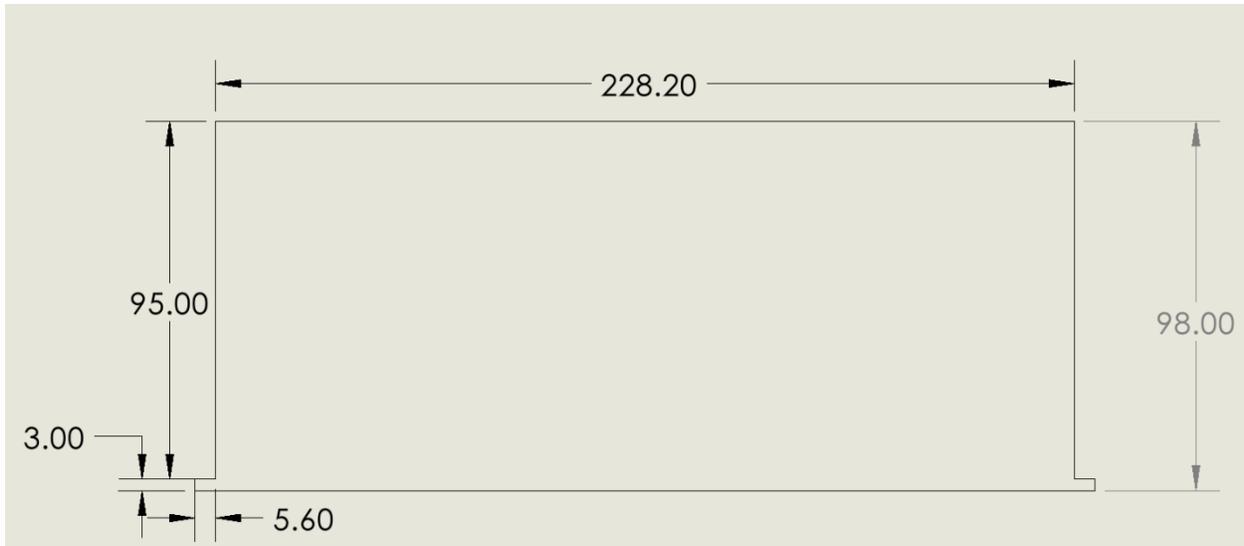


Figure 8: Updated dimensions of CubeSat

Altering the CubeSat size in this manner also addressed a CSD requirement. The maximum size was noted to be a dynamic envelope, meaning that the CubeSat must remain within these dimensions under all conditions, including thermal and vibratory. As the vibration could result in deformation, sizing the CubeSat below the maximum dimensions provided a margin of safety in case the deformation was severe.

With this updated cross-section, a wall thickness of 3 mm was chosen based on the tab thickness. The overall length chosen was the maximum allowed, 366 mm. This resulted in a hollow shell into which the selected components were placed. Determining component placement was a complicated issue. Based on the chosen orbit, two faces of the CubeSat were already reserved: the upper face for the primary solar panel, and the lower face for the antenna. As the top face always approximately faced the Sun, it was the optimal place for the primary solar panel. It was decided to locate the antenna on this bottom face, which always faced approximately away from the Sun,

as it would maintain a relatively constant position and direction, unlike the side faces, due to CubeSat rotation.

While these components' placements were fixed, other subsystems had additional components to place, with varying requirements, described below in Table 3: Additional . The remaining faces, which consisted of two 2U faces and two 3U faces, were all ram-facing and rotating due to the spinning nature of the CubeSat. Additionally, the 3-unit faces were partially obstructed due to the tabs.

Table 3: Additional Component Placement Requirements

Component	Requirement
Neutral Ion Mass Spectrometer (NIMS)	Aperture facing a ram face
Propulsion system	Exhaust facing a ram face
Linear Accelerometer and Rate Gyros	Close to the center of mass
Fine Sun sensors	Facing the Sun

Based on these requirements and the available space, it was decided to locate the NIMS instrument on a 1U by 2U face, the front relative to the dispenser. The propulsion system was located opposite this, facing the rear face of the dispenser. These are the two largest and heaviest components and placing them opposite as such helps maintain the center of mass within the constraints. The remaining components were distributed in the middle of the satellite.

3.3.1 Material Models

The material chosen for the structure of the satellite was Aluminum 7075-T6, as recommended by the payload specification and previous MQPs [30], [48]. A specific numerical

model for this material derived from previous MQPs was used [48]. The tabular data used for this model is shown below in Table 4 and Figure 9.

Table 4: AL7075-T6 Material Properties

Property	Value	Units
Density	2.81	g/cm ³
Young's Modulus	7.17E10	Pa
Poisson's Ratio	.33	
Bulk Modulus	7.024E10	Pa
Shear Modulus	2.6955E10	Pa

Table of Properties Row 10: Chaboche Kinematic Hardening				
	A	B	C	D
1	Temperature (C)	Yield Stress (MPa)	Material Constant C1 (MPa)	Material Constant γ_1
2	20	503	5324	31.06
3	100	448	6226	73.9
4	400	30	1768	28.68
*				

Figure 9: Chaboche hardening parameters

This material met all required outgassing and structural requirements as defined in the CSD, NASA, and CubeSat specifications.

3.3.2 Summary of Mechanical Design

SolidWorks was used to create the model of the CubeSat and calculate total mass, surface areas, and inertia matrix. The model of the CubeSat was produced taking all component positioning requirements into account, as well as structural rigidity. Another goal was to design a

satellite which could plausibly be constructed. A basis for this design was the example satellite from the CSD user manual in Figure 10.

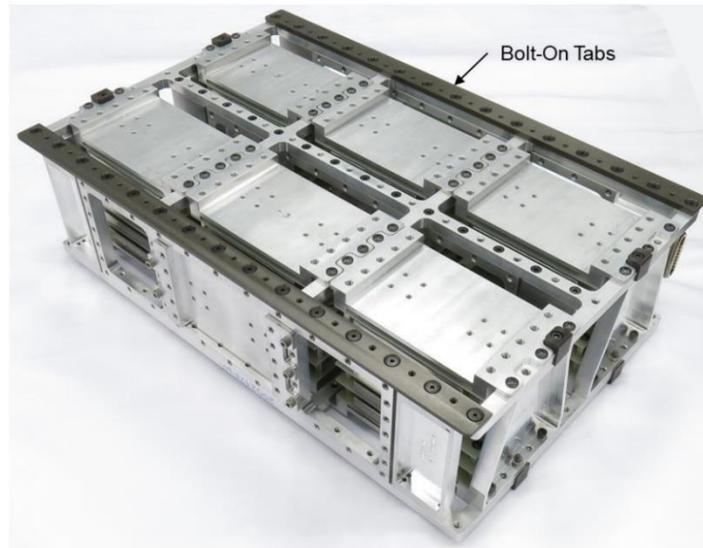


Figure 10: Example 6U Tab CubeSat [30]

This satellite employs a lower and upper frame connected by panels, an idea which was replicated in the mechanical design. However, differing from the example, a solid base plate was created, which served as both the tabs and the primary mounting for the components. A lower frame was attached to this base plate, which was separated from an upper frame by wall components, hollowed to save mass. On the interior of the satellite, three primary compartments were present: the front for the primary instrument, the middle for the electronics stack, transceiver, and inertial sensors, and finally the rear for the propulsion unit. Compartments were separated with additional hollowed wall components, which had allowances for cables to run through. An upper frame was then bolted to the top of the wall components. Single-part side frame components were present on the left and right sides, bolted to the lower frame and base plate. Figure 11 below shows

an exploded view of these primary structural components, including the base plate, upper and lower frames, left and right frames, and the internal structural panels.

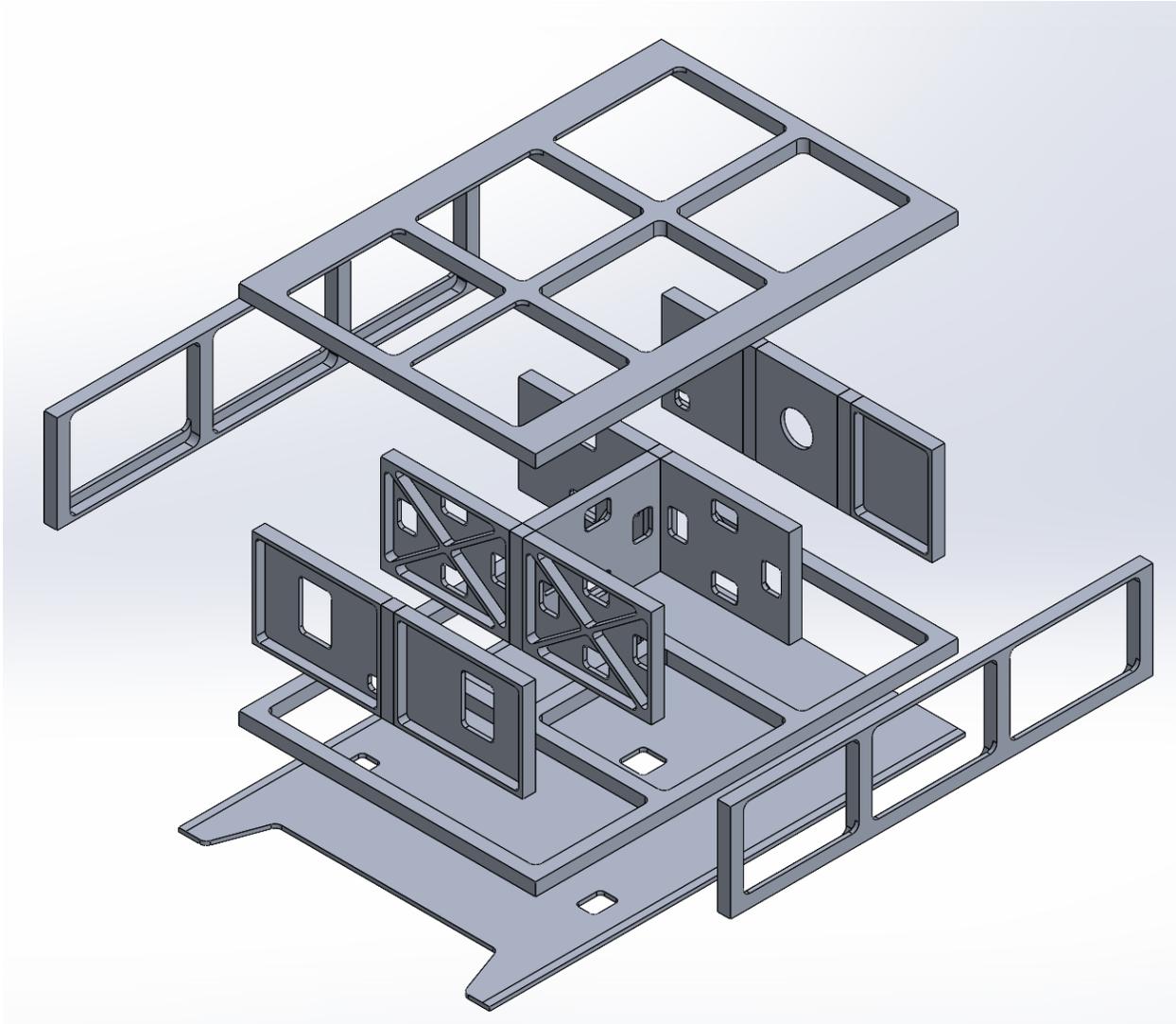


Figure 11: Exploded view of MagSTARS primary structural components

Figure 12 below shows the CubeSat model with all internal components, with the external solar panels transparent:

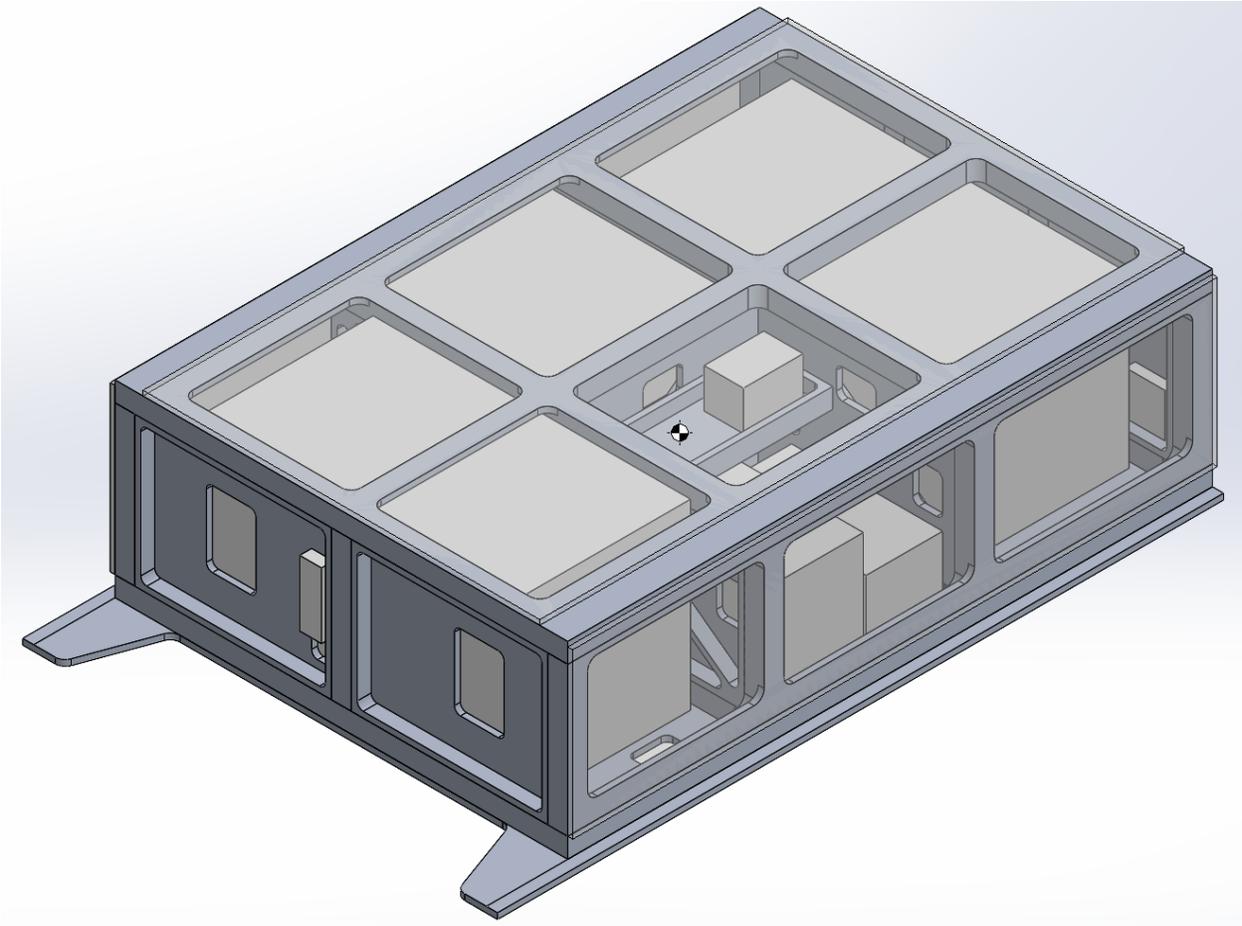


Figure 12: MagSTARS internal design

In this figure, the three primary compartments are evident: the front instrument compartment, the central electronics and computer compartment, divided into two sides, and the rear propulsion compartment. Not shown in this angle are the components mounted to the lower base plate: the antenna and 2U solar panel.

Instrument field of view was also considered, by modeling each field of view with the appropriate cone or pyramid and using SolidWorks to check for collisions. The cutouts on the front panel are thus required to allow the mini-INMS to take in particles. Figure 13 shows these fields of view.

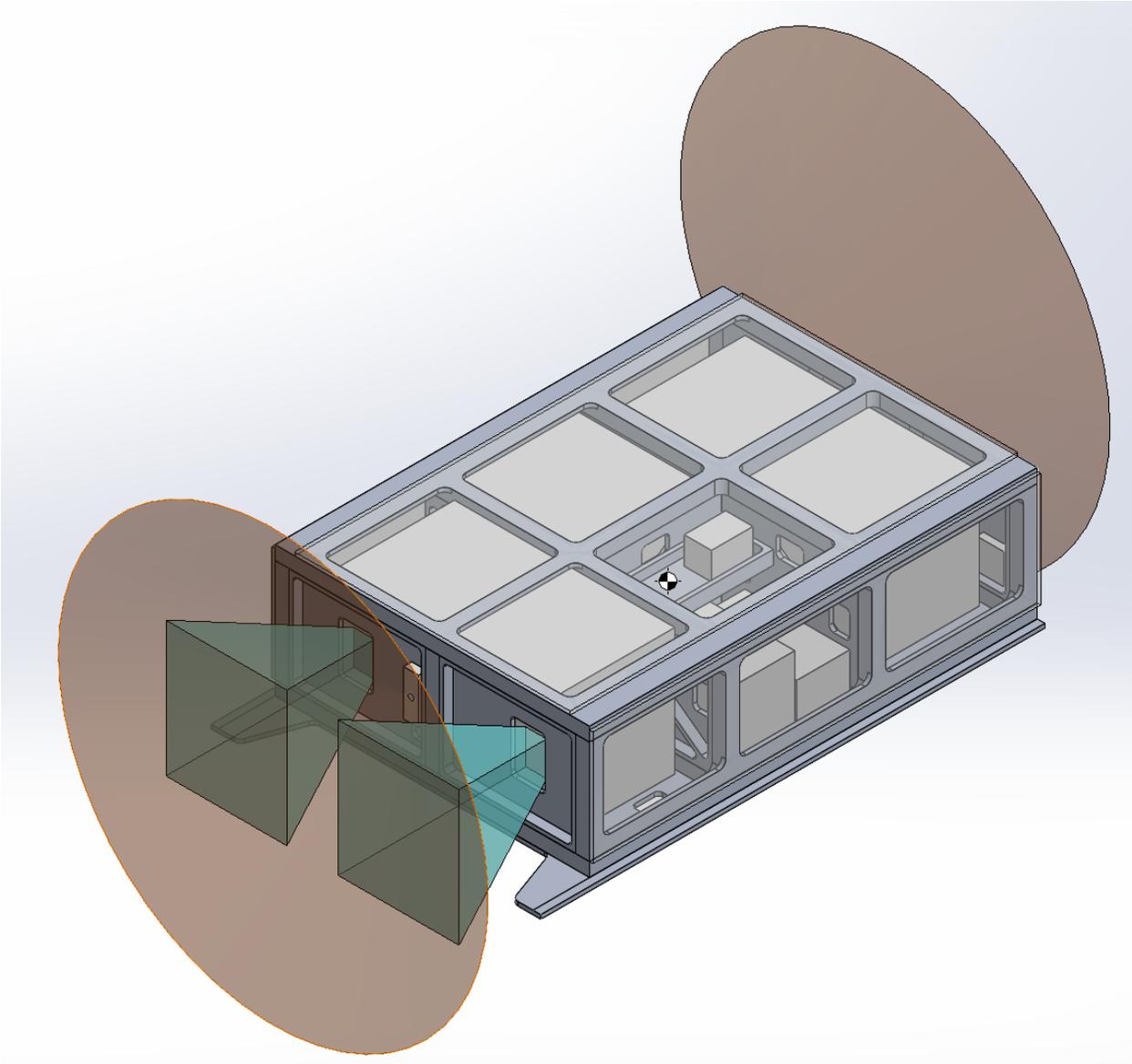


Figure 13: MagSTARS instrument field of view

Because the solar panels have included Sun sensors, separate Sun sensors were only required to be attached on the front and rear faces. It was assumed that the integrated Sun sensors would have no field of view issues, as the solar panels themselves did not. The patch antenna was attached to the bottom, and holes were also provided for these components to theoretically run wiring.

With the component positions and mechanical structure determined, the overall structural characteristics of the CubeSat could be determined. The calculated wet mass was 7334.48 g. For propulsion calculations, a 5% margin was added to this, to make the final wet mass 7701.20 g. With the 258 g fuel mass discussed in Section 4, the dry mass was calculated to be 7443.2 g. SolidWorks also provided the inertia matrix, shown in Figure 14, in kg-m²:

Moments of inertia: (kilograms * square meters)		
Taken at the center of mass and aligned with the output coordinate system.		
Lxx = 0.09716889	Lxy = -0.00037538	Lxz = -0.00030636
Lyx = -0.00037538	Lyy = 0.12113327	Lyz = 0.00055883
Lzx = -0.00030636	Lzy = 0.00055883	Lzz = 0.07737322

Figure 14: MagSTARS inertia matrix

Of additional interest were the cross-sectional surface areas in each axis. Using SolidWorks, these were calculated to be 33,202 mm² on the 1U by 3U face, 23873.95 mm² on the 1U by 2U face, and 80056.834 mm² on 2U by 3U face. These areas were used in Section 4 for drag calculations.

3.4 Mechanical Analysis

For the mechanical analysis, individual components were considered rigid. The primary reason for this is that structural characteristics of each component were not available, and often exact locations of attachment points were similarly unavailable. Because of this, simplified models (rectangular prisms representing a bounding box) were used when conducting analysis in ANSYS. These models were set to the correct mass by utilizing custom materials for each part. This approach does have several limitations. First, this approach assumes that the mass in each component is uniformly distributed in the representative prism. This would not be the case in an actual satellite. However, even if a more detailed 3D model was acquired, it may not accurately

model the mass distribution either. Thus, this approach was considered to be acceptable for the team's purposes.

Second, the components are not truly rigid models, and considering them to be introduces more error into the results. This was partially mitigated by avoiding attaching rigid components to other rigid components, with the exception of the power distribution unit and electronics stack. Attaching rigid bodies to other rigid bodies raises an error in ANSYS and could lead to inaccurate results. However, the exact deformation and position under load of these components was not the primary focus of the mechanical analysis, so this was judged as acceptable. Furthermore, 3 mm of clearance was generally maintained around each rigid component, with the exception of the primary attachment point. This ensures that rigid components are not acting as supports across multiple faces themselves.

Finally, as this project assumes extra propulsion tanks as described in Section 4.2, a representative cuboid was determined to have three times the volume of one propulsion unit and the mass as described in Section 4.2. As only the tank portion would have to be increased, and not the thruster itself or any electrical hardware, this volume is likely an overestimate. This volume had the same height, two times the width, and 1.5 times the depth as the single thruster unit, as it was assumed there would be some flexibility when adding tanks due to lack of documentation.

3.4.1 Analysis Setup

To verify that the subsystem was meeting each performance requirement, several types of analysis were required. To conduct these analyses, the CAD model of the CubeSat assembly was imported into ANSYS. Each component was assigned the correct material and defined as rigid or flexible. Due to software limitations, a separate material had to be defined for each component in

order to correctly account for the mass. This material was defined as having a density equal to the known mass of the component divided by the volume of the prism, resulting in a correct overall mass. Details of these materials are described in Appendix A.

ANSYS then required each connection between bodies to be defined. Most connections were defined as bonded, simulating a tight bolted connection between components. The rigid elements in the electronics stack had their behavior changed to be asymmetric due to their rigid nature. The mini-INMS instrument and the propulsion unit, which contacted both the bottom and top faces, had their top contacts set to frictional in order to prevent the rigid bodies from adding an excessive amount of structural support. The vertical bars, which are bolted horizontally to the interior walls, were modeled as frictional with the base and roof plate, as they are not directly bolted. All other behaviors were allowed to be controlled by the program.

The final setup step was to create a finite-element mesh of the model. Mesh size selection can greatly influence Finite-Element analysis results. Through a trial-and-error process, mesh size was started at small values (~1 mm). These results resulted in unacceptably long computation times and RAM usage, especially when considering the number of simulations to perform: three orientations with two different vibration spectra. Eventually, the default element size was set to 3 mm, and other settings left at default. This resulted in the meshed model shown below in Figure 15, which completed one analysis cycle in approximately ten to fifteen minutes.

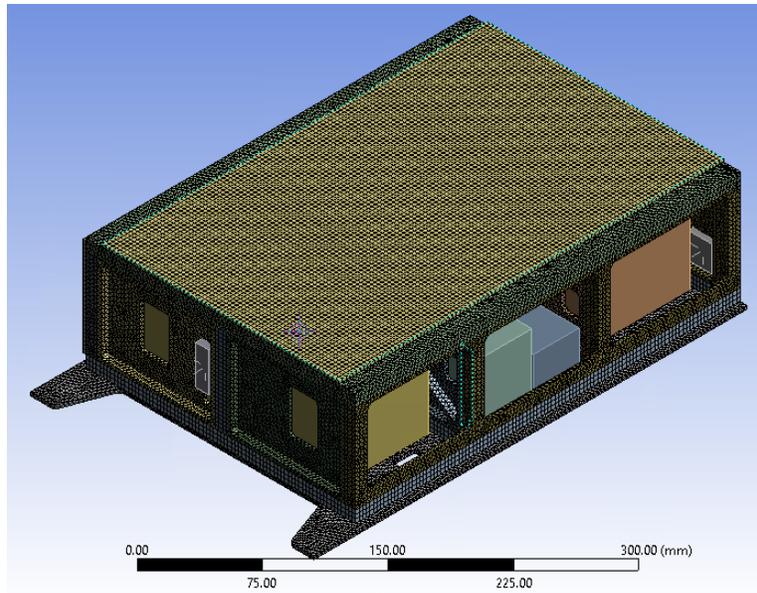


Figure 15: Meshed model of CubeSat

This mesh refinement technique of successively reducing the element size is the simplest, but it has drawbacks. The mesh is approximately the same size everywhere, both in areas of interest and less important areas [49]. This could lead to wasted computation time, or important areas not being treated appropriately. In this case, every area is equally important, so this drawback is not significant. With these setup steps completed, analysis could now begin.

3.4.2 Modal Analysis

The first form of structural analysis conducted was modal analysis. Structures respond more significantly to vibration at specific frequencies, resulting in shapes called modes. Determining these critical frequencies is commonly required for a variety of reasons, especially to determining that an outside periodic excitation does not cause resonance and lead to structural failure [49]. Given the high-vibration environment that occurs during a spacecraft launch, the

SpaceX payload specification recommends that the lowest mode be no less than 40 Hz in order to avoid any potential damage to the CubeSat [10].

Several steps were required to setup the modal analysis. First, the range to search for modes and the number of modes to find was required. Based on information found in the payload specification guide, it is good practice to set the maximum search frequency to 1.5 times the maximum frequency in the random vibration excitation [30]. This frequency in both the SpaceX and NASA vibration spectra is 2000 Hz, thus the range searched for modes was 20-3000 Hz. Ninety-nine modes were searched for, as in previous MQPs, to ensure all modes were found [23]. This has the potential drawback of finding less significant modes.

Two more prerequisites remained. An important factor in determining modes is properly defining supports. Based on the PSC specification of clamped tabs, whose purpose is to provide a well-defined load path from the CSD to the CubeSat, the top and bottom faces of the tabs were modeled as fixed supports [30]. The clamps are not truly fixed relative to the satellite, but this approximation is appropriate based on the CSD payload specification. The tabs are modeled as fixed in all three axes, constrained by the clamps in the vertical direction, the CSD walls in the lateral direction, and the CSD back and front plate in the longitudinal direction.

3.4.3 Random Vibration Analysis

Random vibration analysis involves subjecting a structure to a specific spectrum of vibration and determining the response. This is very important for a spacecraft due to the wide variety of vibration environments they encounter during launch and operation. The spacecraft must maintain structural integrity and remain in an acceptable shape with minimal deformation. In the case of this project, the CSD specification imposed an additional requirement: the loading on the

tabs must not exceed 3559 N. The advantage of the CSD is that providing this requirement is met, there is no maximum mass, providing room for this project's heavy main instrument and propulsion system.

NASA defines a testing spectrum in the General Environmental Verification Standards (GEVS) for spacecraft shown below in Figure 16 [50]. As MagSTARS was a small spacecraft, it had to be subjected to the highest vibration spectrum. SpaceX also defines a separate vibratory spectrum the satellite must be subjected to, as shown in Figure 17 below. Comparing the two spectra results in the following the graph shown in Figure 18. It is apparent from Figure 18 that the two spectra intersect at around 1000 Hz. This means that each individual spectrum must be tested, as neither strictly contains the other. Despite the SpaceX spectrum having much less total root mean square acceleration (5.13 Grms compared to 14.1 Grms), it was still necessary to independently test both spectra to ensure no significant modes existed at the portion where the SpaceX curve is higher.

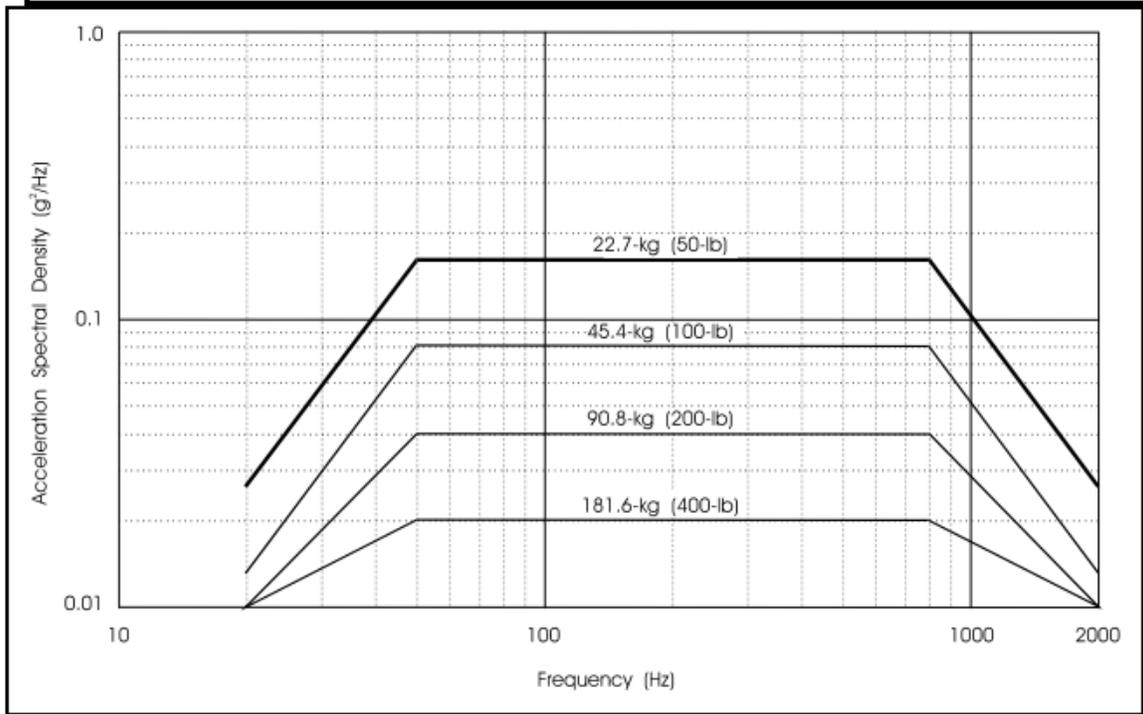


Figure 16: NASA GEVS random vibration spectrum [50]

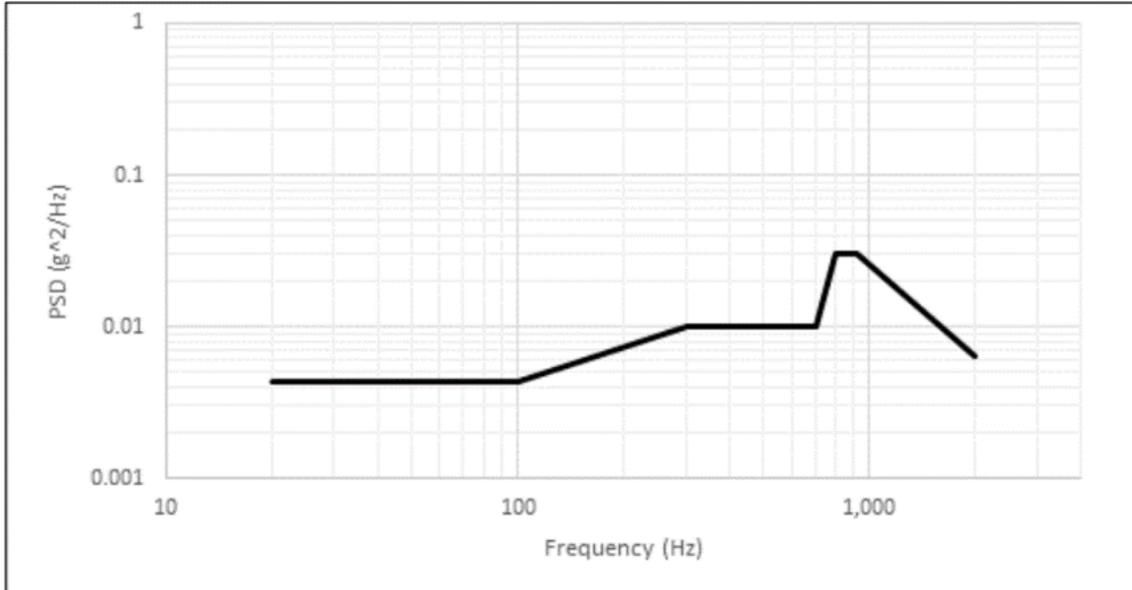


Figure 17: SpaceX random vibration spectrum [10]

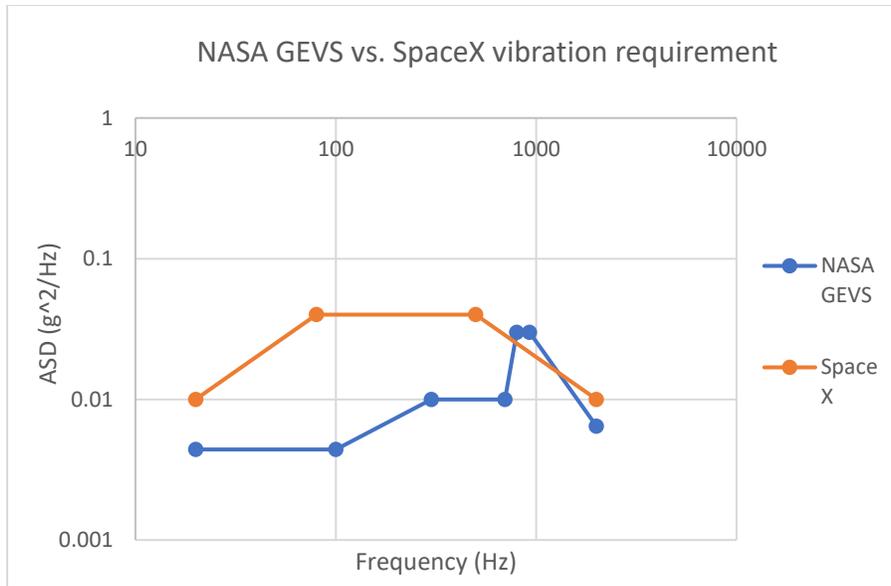


Figure 18: Comparison of vibration spectra

To conduct the random vibration analysis, the vibratory spectrum was input into ANSYS as a PSD Gee acceleration. The rest of the setup could be expedited by using the previously computed vibrational modes module as an input to the random vibration module. With the addition of a force reaction probe object, the tab loading, deformation, and equivalent stress were determined.

In Figure 19, the overall structure of the ANSYS setup is shown, with the results of the modal analysis being fed into each of the random vibration spectra. As the material properties, geometry, and supports are the same in each case, the Engineering Data, Geometry, and Model sections can be carried over to each.

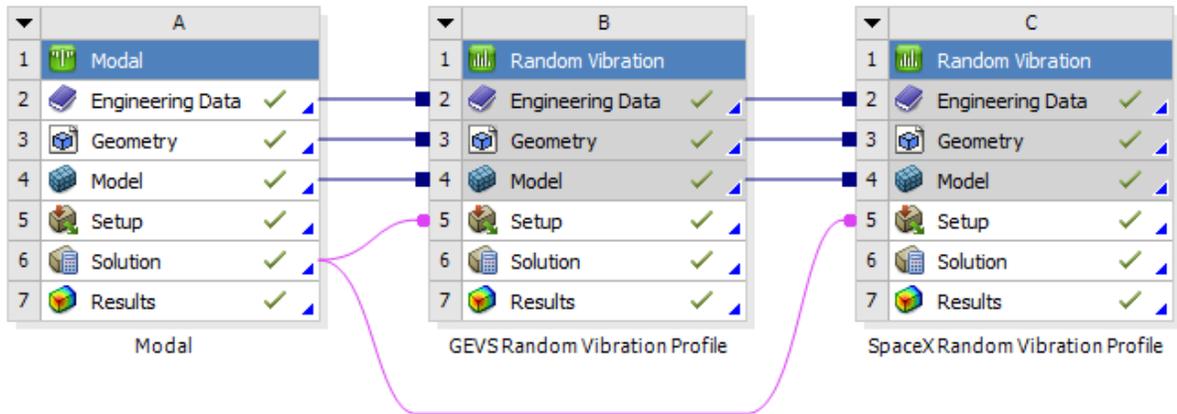


Figure 19: ANSYS block diagram

With all setup complete in SolidWorks and ANSYS, results were calculated.

3.4.4 Results

The center of mass of the satellite was determined using SolidWorks and was measured relative to the coordinate system defined in the payload specification: a point on the rear face of the satellite, at the lateral plane of symmetry, at the lower plane of the tab plate, labelled as “origin” in Figure 20 below.

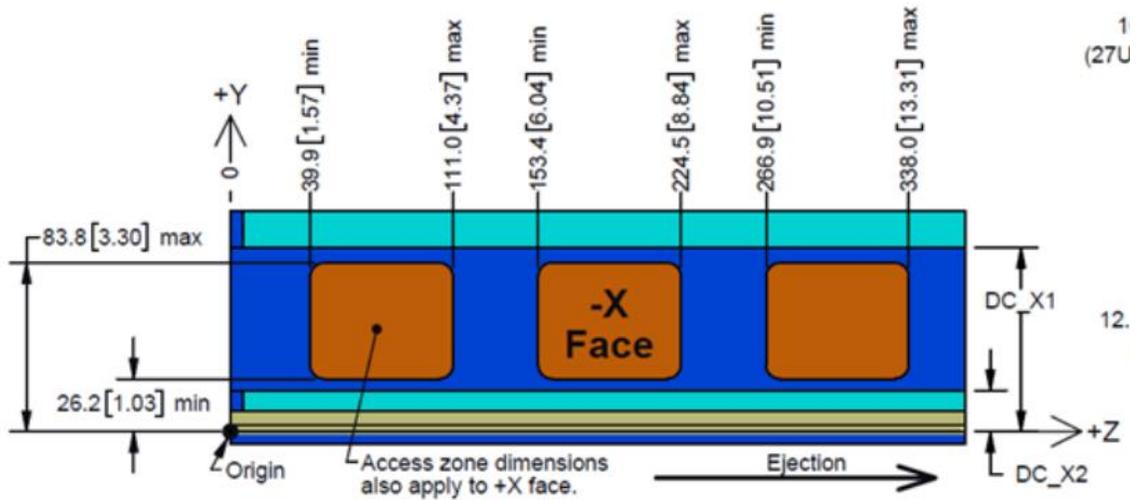


Figure 20: CubeSat-centric coordinate system [30]

The location of the center of mass compared to the requirements is shown below in Table

5:

Table 5: Center of Mass of MagSTARS

Property	Requirement	Measured Full Fuel	Measured Empty Fuel
Center of Mass, X	-40 to 40 mm	-1.8843 mm	-1.9562 mm
Center of Mass, Y	10 to 70 mm	45.078 mm	45.0611 mm
Center of Mass, Z	133 to 233 mm	152.3269 mm	155.9284 mm

The center of mass is within the required envelope. Because the satellite will only be contained within the CSD while containing at its full wet mass, it is not necessary to verify that the center of mass will remain within this envelope with empty fuel. This value has been included for reference, however.

A venting analysis was also required. The total ventable volume (i.e. empty space within the CubeSat) divided by the total venting area (i.e. open wall portions) must be no greater than 2000 in as defined by the CubeSat specification [28]. Using SolidWorks to determine the internal volume, excepting space filled by components, the empty volume was determined to be 2.44×10^6 mm³. The ventable wall area was determined to be 3186.52 mm², based on the IMNS field of view cutouts, the Sun sensor cutouts, the propulsion cutout, and the bottom venting cutouts. This results in a ventable volume per venting area of 765.995 mm, or 30.1572 in, which is well below the maximum of 2000 in. The internal volume disregarding the components was also evaluated, as the components were represented as simple cuboids and significant air may be present within these bounds. This resulted in a volume of 1.63×10^7 mm³. which results in a ventable volume per venting area of 5105.87 mm, or 210.02 in, again well below the maximum of 2000 in.

From the modal analysis, the lowest significant mode was found at 1859.8 Hz, involving the base plate deforming due to the mass of the mini-INMS instrument. The only other mode below 2000 Hz was at 1890.9 Hz, relating to the mini-INMS instrument once more. All modes found between 20 and 3000 Hz are shown below in Figure 21.

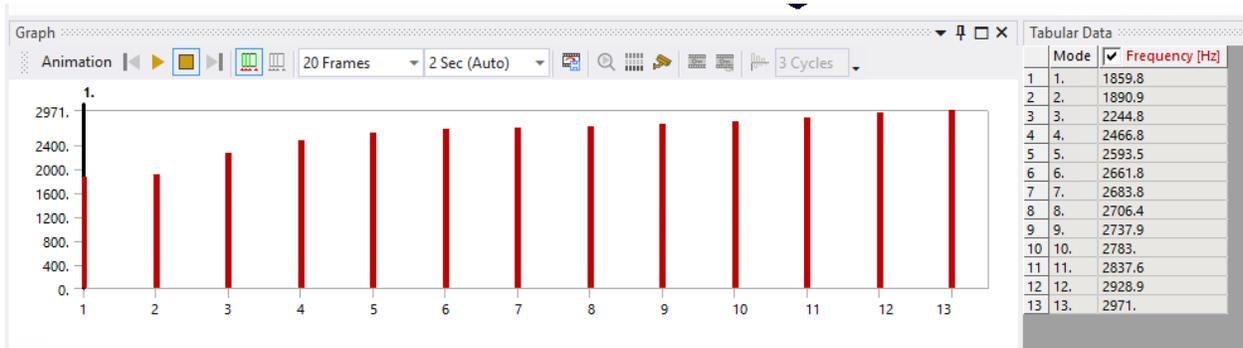


Figure 21: MagSTARS calculated vibrational modes

Thus, the modal requirement of no modes below 40 Hz was met and exceeded. This modal result was then fed to the random vibration analysis for the next step.

From the NASA GEVS vibratory spectrum, simultaneous 3-axis random vibration excitation results are as follow in Table 6:

Table 6: NASA GEVS Vibratory Response

Result Type	Requirement	Determined Value
Tab Loading	3559 N	Y-axis: 2432.0 N
Tab Loading with static launch load 6 G	3559 N	2885.29 N
Tab Loading with static launch load 12.9 G	3559 N	3406.58 N
Maximum deformation	Do not exceed envelope	X-axis: .0059759 mm Y-axis: .0072047 mm Z-axis: .011031 mm
Maximum stress	503 MPa	32.805 MPa

It can be seen that each requirement is met, including maximum stress with a safety factor of 15.33. The location of the stress is primarily in the base plate, as seen in Figure 22.

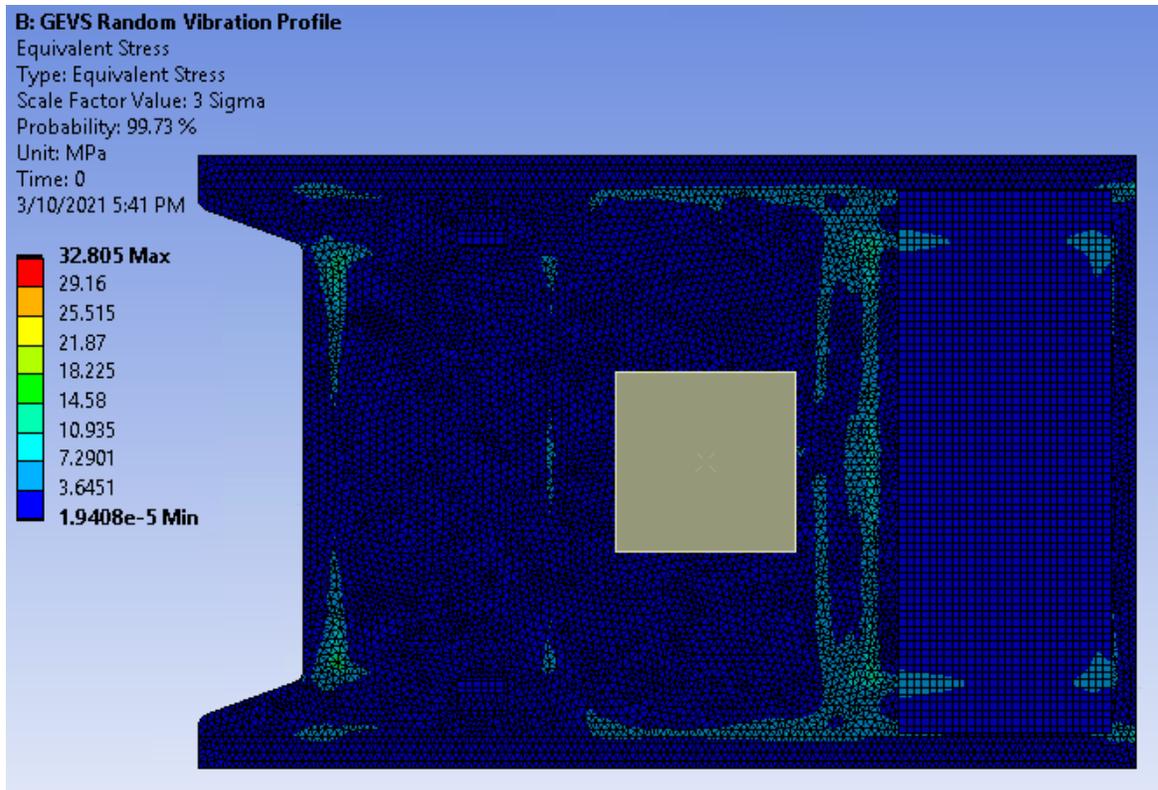


Figure 22: Stress in MagSTARS base plate

For the tab loading, the Y-axis loading was most significant, based on the 3559 N maximum loading relating to the springs used for clamping. Due to the satellite being constrained by the walls of the CSD in the X and Z axis independent of the tabs, it was assumed that only the Y-axis tab loading was significant. Also listed is the Y-axis tab loading when considering a maximum 6 G and 12.9 G launch acceleration, based on the SpaceX rideshare user manual and Falcon 9 manual, which could occur depending on the orientation of the CSD relative to the launch vehicle [10], [51]. This static loading represents only a small increase relative to the vibrational loading, so again, the maximum tab loading is not exceeded.

In addition, the listed deformations all do not make the satellite exceed the required envelope. In each direction, the only components close to the maximum dynamic envelope are the

tabs themselves, which meet the strict size and position requirements, and deform the least amount. The upper frame has significant clearance to the left, right, and front, and the rear face is an acceptable contact zone. Plots of these deformations, in the Y, X, and Z axis respectively, follow in Figure 23, Figure 24, and Figure 25.

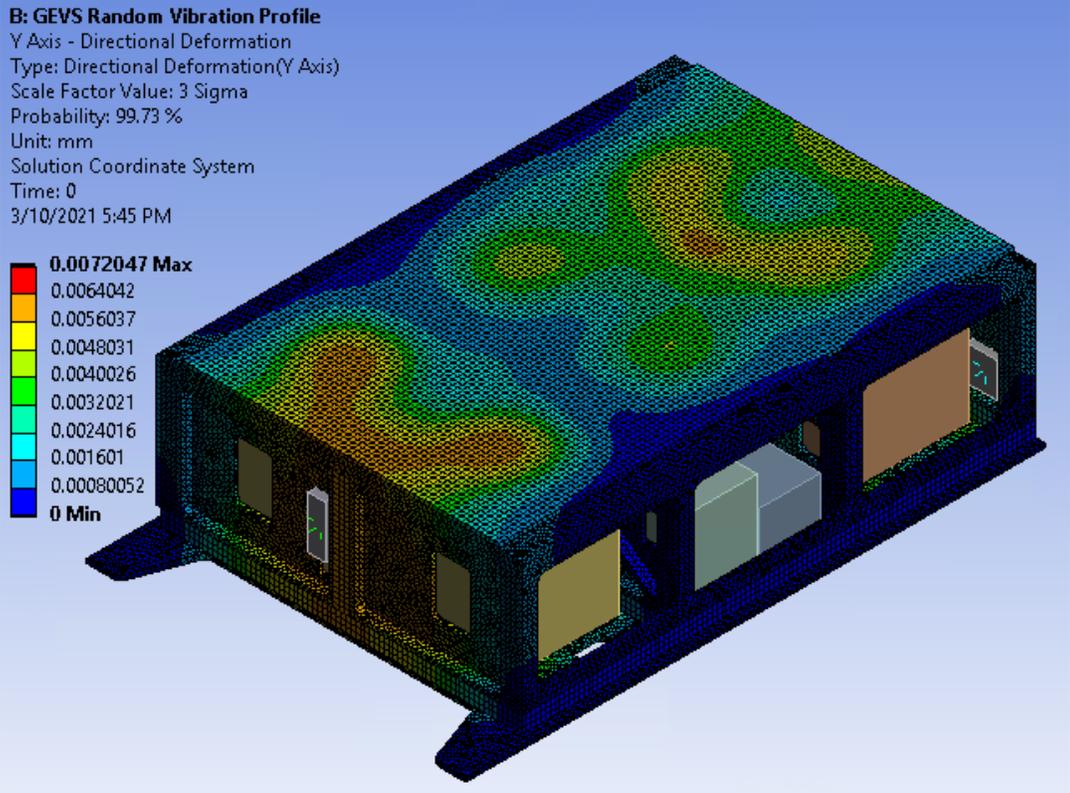


Figure 23: Y-axis GEVS deformation

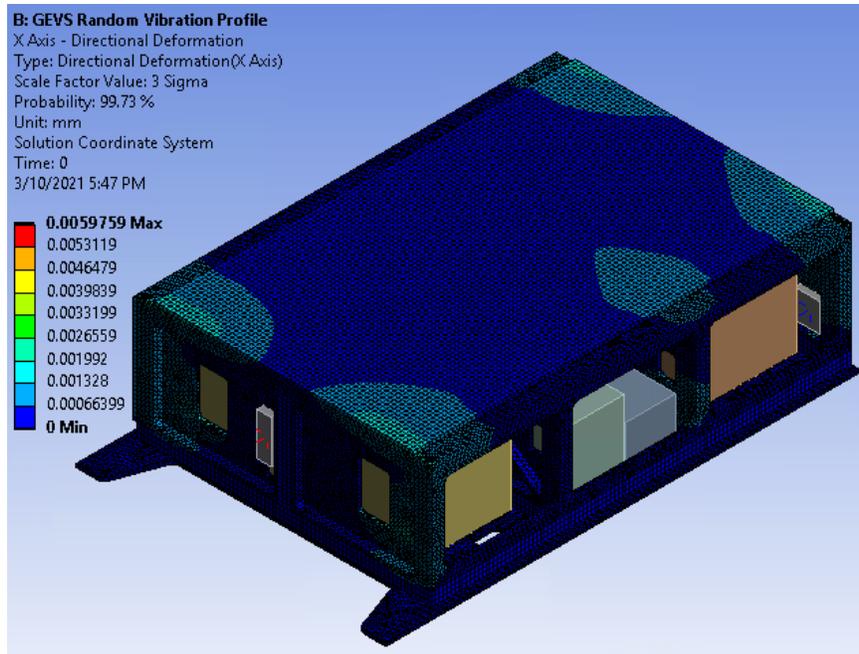


Figure 24: X-axis GEVS deformation

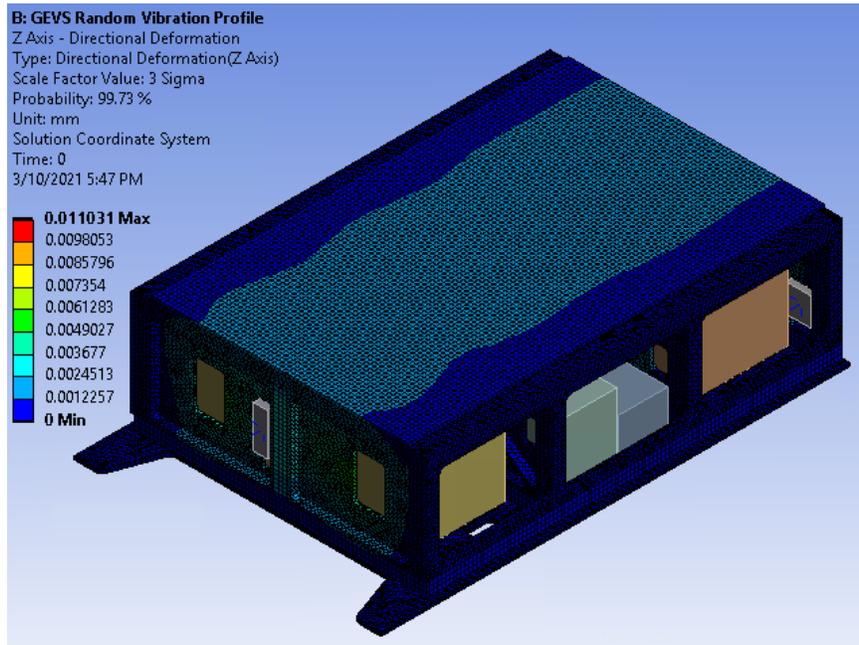


Figure 25: Z-axis GEVS deformation

From these figures, it is apparent that most of the deformation occurs on the center, top, and front of the CubeSat. Each of these areas has significant clearance from the CSD: 9.2 mm on the top, including the solar panel, 41 mm on the front, and 6.5 mm on the bottom. Thus, deformation due to random vibration is not a concern for MagSTARS.

From the SpaceX vibration spectrum, results are as follow:

Table 7: SpaceX Vibratory Response

Result Type	Requirement	Determined Value
Tab Loading	3559 N	Y-axis: 1492.7 N
Tab Loading with static launch load 6 G	3559 N	1946.0 N
Tab Loading with static launch load 12.9 G	3559 N	2467.3 N
Maximum deformation	Do not exceed envelope	X-axis: 0.0040806 mm Y-axis: 0.0049295 mm Z-axis: 0.0076764 mm
Maximum stress	503 MPa	22.636 MPa

In this case, maximum stress had a safety factor of 22.22. This vibration spectrum resulted in lower tab loading, deformation, and stress compared to the NASA loading. As before, the deformation does not exceed the maximum dynamic envelope. As every result was lower than the GEVS results, individual deformation images are omitted, but can be found in Appendix B.

As shown above, the CubeSat meets or exceeds each mechanical requirement defined by the dispenser datasheet, NASA, SpaceX, and the CubeSat specification. Furthermore, due to the clamping of the tabs, these results can be considered to have a higher degree of accuracy than results from a rail-based CubeSat, which might have additional room to vibrate and cause damage.

4 Propulsion Analysis

This chapter discusses the propulsion subsystem of MagSTARS, and the process to select a propulsion system. It also outlines the setup and analysis of an optimal orbit for the science phase of MagSTARS' mission.

4.1 Propulsion Overview

The primary roles of the propulsion subsystem were to perform orbital insertion and orbit maintenance. The propulsion system was to provide enough thrust and ΔV to compensate for drag in order to optimize the orbital lifetime of MagSTARS while it completed its science mission.

Two classes of thrusters are typically used for orbital adjustment and maneuvering of satellites: primary thrusters and secondary thrusters. Primary thrusters are used for larger maneuvers of the spacecraft and typically provide enough thrust to perform higher magnitude ΔV maneuvers. Secondary thrusters typically are used to orient the spacecraft in a desired heading [52]. For this mission, MagSTARS only used a primary thruster due to volume and mass constraints discussed in Section 1.5.1.4. This was sufficient, as MagSTARS also had a magnetorquer for attitude control, as described in Section 7.2.1.

There are numerous methods of choosing a primary thruster. In the article *Propulsion for CubeSats*, Kristina Lemmer recommends considering the following: thruster efficiency, specific impulse, total impulse, and impulse density [53]. Efficiency can be difficult to use as an assessment criterion alone because it has many different meanings for different types of propulsion systems. Specific impulse alone does not give any indication of total impulse or time to achieve said impulse. Total impulse depends on the amount of propellant a satellite can carry, and thus, the

same amount of propellant must be used for each system. This is not always the best assessment criteria, because different propulsion systems have different propulsion densities and masses [53].

For this project, the propulsion system was selected primarily based on constraints, such as the amount of power available and overall mass. The following steps were followed when choosing a propulsion system:

1. Make a list of mission maneuvers which require propulsion
2. Determine ΔV budget, fuel consumption, burn time, and total impulse for propulsion functions and drag compensation
3. Determine propulsion system options
4. Estimate key parameter of each propulsion system option
5. Perform trade study on different propulsion system options
 - Consider thrust capability, specific impulse, and total impulse
 - Estimate total mass, volume, and power for each option
6. Establish baseline propulsion system

Steps 1-5 compiled enough information to select a propulsion system for MagSTARS which met all requirements, as listed in Step 6. For example, the required ΔV budget requirement gave an estimate of how much specific impulse would be needed to lower the mass of the propellant. This was helpful while looking at different propulsion system options. The broad categories of propulsion systems are cold gas, electric propulsion (electrothermal, electrostatic, and electrodynamic), and chemical propulsion (monopropellant, bipropellant, and solid propellant). Some of these options are shown in Table 8 with their thrust and specific impulse ranges.

Table 8: Propulsion System Types [54]

Propulsion System Type	Thrust	Specific Impulse (s)
Hydrazine	0.5-30.7 N	200-235
Cold Gas	10 mN-10 N	40-70
Green Propulsion	0.1-27 N	190-250
Pulsed Plasma Thrusters	1-1300 μ N	500-3000
Electrospray Propulsion	10-120 μ N	500-5000
Hall Effect Thrusters	10-50 mN	1000-2000
Ion Engines	1-10 mN	1000-3500
Solar Sails	0.25-0.6 mN	N/A

These types of parameters were used to determine which propulsion system would be best in the interest of optimizing the lifetime of MagSTARS in the science phase of its mission. The requirements found in STK and those that came from other subsystems, such as power, were also considered while choosing a propulsion system.

4.2 Thruster Selection

The selection of the primary propulsion system for MagSTARS was done through a trade study. This consisted of research on each type of system (listed in Table 8), an initial down selection of thrusters within each type of system based on preliminary calculations for the

spacecraft, and then further narrowing options and adjusting other parameters to best fit the remaining options in an iterative process.

The initial down-selection was done by investigating the typical range of specific impulse for each type of thruster. This was done with the knowledge that the value would need to be as high as possible in order to decrease the propellant mass, as given by the Rocket Equation:

$$\Delta V = -c \ln \left(\frac{m_f}{m_i} \right) \quad (1)$$

where ΔV is the change in velocity during a burn in m/s, m_f is the final mass of the satellite in kg, m_i is the initial, or wet, mass of the satellite in kg, and c is the effective exhaust velocity in m/s which is given by the equation:

$$c = g Isp \quad (2)$$

where g is the acceleration of gravity (9.81 m/s^2), and Isp is the thruster's specific impulse in seconds.

In order to achieve a low propellant mass, a rough estimate of the required specific impulse was found using the estimated ΔV requirement for the maneuvers requiring burns. This showed that specific impulses of at least 300 seconds or higher was optimal. In order to find inclination values, the initial and final orbits were entered into STK as Sun-synchronous and their altitudes were specified. The initial orbit had an inclination between approximately 97.41° and 97.89° , and the final orbit had an inclination between 96.33° and 96.68° . Using these values in the MATLAB script in Appendix C, the maximum required ΔV budget to reach the science orbit was found to be approximately 355 m/s. The inclination change from the initial orbit to the final orbit occurred during the first impulse, rather than the second, as this resulted in a lower required ΔV budget. As

discussed in Section 4.3.2, it was later decided not to do this inclination change, however, the propulsion systems were first analyzed with an inclination change.

Considering initial power consumption and power generation estimates from the power subsystem, the propulsion subsystem would have approximately 8 W of available power. Using this power budget estimation, the specific impulse range constraint, and the restart capability requirement of the thrusters for MagSTARS, the initial down selection of the propulsion options was performed. The remaining options and their performance specifications are listed in Table 9. Parameters in this table were used to give approximations of mass and volume of the propulsion subsystem to the mechanical subsystem. The mechanical subsystem initially allocated half of a 1U space to the propulsion subsystem, so each of the thrusters met this requirement, but while the micro-PPT would only use a small portion of this space, the BET-100 and the SiEPS had much lower wet masses. The BET-100 and the SiEPS were both within the power budget and had high specific impulse values, however, their thrust capabilities were extremely small. Thus, the micro-PPT thruster was kept in consideration, despite initial power allocation estimations, due to its higher thrust capability and its reasonable specific impulse value. However, it was noted that the BET-100 had a much higher total impulse capability than the micro-PPT. The BET-100 and the SiEPS were similar overall, however the BET-100 had a higher specific impulse, while the SiEPS used less power, had less wet mass, and was approximately half the size of the BET-100. Originally, the BET-1mN was not considered due to power constraints in conjunction with its dimensions making it unable to fit into a half of 1U space.

Table 9: First Down-Selection of Thruster Options and Their Parameters [22], [51], [52], [53]

Thruster	Manufacturer	Thrust	Isp	Power	Impulse	Wet Mass	Volume
BET-100	Busek	100 μ N	2300 s	5 W	175 Ns	0.55 kg	9x9x4 cm
Micro-PPT	Busek	0.14 mN	500 s	12.5 W	10-80 μ Ns	3.8 kg	3.25x1.25x1.25 cm
SiEPS	MIT	100 μ N	1200 s	1.5 W	-	0.1 kg	9x9.6x2.1 cm or .2U
BET-1mN	Busek	0.7 mN	800 s	15 W	675 Ns	1.236 kg	8.5x8.5x6 cm

Once set up, the STK full mission scenario described in Section 4.3.2 was used to calculate a more accurate required ΔV budget. This was done by running the Mission Control Sequence in STK and applying changes from the target sequences for lowering the perigee and apogee altitudes to their respective maneuvers. The total required ΔV from this analysis (approximately 230 m/s without an inclination change) was compared to that previously found in MATLAB for confirmation (approximately 239.8 m/s without an inclination change). Using the more accurate required ΔV value from STK, (1), and the two equations below, the remaining thruster options from the first down-selection were tested using their specific impulse values. Total impulse is given by:

$$I = g Isp m_p \quad (3)$$

where I is the total impulse required in N-sec, and m_p is the mass of propellant in kg used in the ΔV burn. Propellant mass is given by:

$$m_p = m_i - m_f \quad (4)$$

Available total impulse and available propellant mass were tested by checking if the resulting required propellant mass and required total impulse were higher than the propellant mass and total impulse provided by each thruster. To do this, Eq. (1) – Eq. (4) were used with input parameters from each thruster and the total required ΔV described above to calculate the required total impulse and required propellant mass to achieve the required ΔV using each thruster. The results of required propellant mass and total impulse for each thruster in Table 9 are shown in Table 10. These results used an estimated wet mass of 7.50 kg in (1), Eq. (3), and Eq. (4).

Table 10: First Down-Selection of Thrusters' Required Propellant Mass and Total Impulse

Thruster	Required Propellant Mass (g)	Required Total Impulse (Ns)
BET-100	76.14	1716.23
Micro-PPT	343.91	1685.14
SiEPS	145.26	1708.24
BET-1mN	216.83	1699.94

At this point, it was found that no single standard thruster would have enough total impulse to complete the mission, as no thruster had a total impulse high enough to achieve the required ΔV . However, the power budget was updated and allotted the propulsion subsystem up to 15 W of power at a time during burns. The mechanical subsystem also updated the allocated space for the propulsion system to nearly 2U. This allowed for larger, higher power thruster options with higher total impulse capabilities to now be considered, such as the BET-1mN described in Table 10. This increase in available power also allowed for the consideration of multiple thrusters and, for thrusters with customizable tanks, increase in tank volume was also analyzed. Electrospray thrusters with customizable tanks typically use pressure-regulated flow, or an active feed system,

while passive feed systems are only expandable to a point, as they rely on capillary action. In addition to adding thrusters with higher total impulse capabilities, the required total impulse was also decreased at this time. As described in Section 4.3.1, the mission was further analyzed in order to lower required ΔV values (and thus lower required total impulse values) and extend MagSTARS' lifetime. The BET-1mN has a customizable tank and a higher total impulse capability than the remaining thruster options, thus it was considered further.

The wet mass of the BET-1mN shown in Table 9 had to be calculated due to lack of documentation on the thruster. The propellant mass per tank was calculated using (3) above with inputs of impulse and specific impulse from Table 9. It was found that there was 0.086 kg of propellant per tank in the BET-1mN. The dry mass was documented as 1.15 kg [53], [54]. Adding the propellant mass per tank to this dry mass, it was found that the wet mass of the system was 1.236 kg. Using the propellant mass in (5) and (1), it was found that the thruster did not provide sufficient ΔV for even the orbital insertion maneuvers—as described in Section 4.3.3, 0.221 kg of propellant mass was needed for orbital insertion. The capabilities of using a two-thruster system or an increased tank volume were investigated. The power used by a single BET-1mN thruster is 15W, which is the propulsion subsystem's maximum power allocation during burns. This meant two of these thrusters could not run at the same time while operating at maximum power. Therefore, it was decided to expand the tank volume, as this would not affect power consumption.

Based on the fuel estimation for orbital insertion, increasing the tank volume by 2.6 times the original volume would suffice. However, it was decided to increase the tank volume by 3 times to provide a safety margin. This expansion factor was then used in the code in Appendix D to determine the ΔV budget with this expanded tank, approximately 267 m/s. As shown in this code, it was assumed that the mass of the original tank volume was approximately 10% of the BET-1mN

thruster dry mass (0.115 kg), again due to lack of documentation. Thus, the wet mass for the original tank volume was 0.201 kg. The calculated ΔV budget was used to determine how much ΔV remained after the Science Leg orbit insertion to be used for stationkeeping and worst-case scenarios, described in Section 4.3.3.

By increasing the tank wet mass by a factor of 3 and using the tank dry mass assumption previously explained, the thruster would have a dry mass of approximately 1.38 kg and a wet mass of approximately 1.638 kg with 0.258 kg of propellant available. The specific impulse and thrust of the system would remain at 800 s and 0.7 mN, respectively, and 15 W of power would be used during burns. The impulse would increase by a factor of 3 to be approximately 2025 Ns. Using the assumptions described in Section 3.4, the volume was estimated to be at most $8.5 \times 10^9 \text{ cm}^3$ with use of a tank expanded by 3 times its original volume.

4.3 Orbital Analysis

Orbital analysis for MagSTARS was done using STK. The main objectives of this analysis were to model the initial orbit, transfer trajectory, and final orbit; calculate key parameters, such as required ΔV and burn time; and optimize the Science Leg duration by trading remaining propellant, drag at the lowest altitude, and orbit shape. This section describes the complete orbital analysis.

4.3.1 Science Leg Orbit

There were many apogee and perigee altitude combinations to consider for the Science Leg of the mission when determining which orbit would both optimize lifetime and support data collection in the F layer, as described in Section 2.2. Originally, MagSTARS was planned to be on a 200 km Sun-synchronous Science Phase orbit. However, as shown by Table 11, this orbit

would not offer a long lifetime for the satellite. Additionally, much more propellant mass would be needed for a Science Leg orbit this low, as it would take more fuel to maneuver from the deployment altitude to the Science Leg orbit, and de-orbit would happen more quickly than at higher altitudes, thus requiring more stationkeeping burns. As mentioned in Section 4.2, the mission was further analyzed once MagSTARS' required total impulse was determined to be too high using the required ΔV to lower to this 200 km Sun-synchronous orbit. By inserting a spacecraft with the same Keplerian elements as the Science Leg orbit into STK and using the same initial state coefficients and areas for drag and Sun exposure as in Figure 26, the Satellite Lifetime Tool was used to compute the lifetime of MagSTARS on the Science Leg orbit using the most recent atmospheric density model, DTM 2012. Four orbits were analyzed in the F layer to determine which would be best for MagSTARS and the remaining thruster options at the time. These orbits and their estimated lifetimes are shown below in Table 11.

Table 11: Lifetime of Various Potential Final Orbits of MagSTARS Without Stationkeeping

Perigee Altitude (km)	Apogee Altitude (km)	Estimated Lifetime Without Stationkeeping (days)
200	200	6
180	400	37
200	400	62
250	400	191

The lower orbits required more propellant to maneuver to the Science Leg orbit. For instance, based on initial estimates, a 180 km by 400 km orbit would require approximately 0.19 kg of propellant to arrive at the Science Leg mission, while a 250 km by 400 km orbit would

require approximately 0.169 kg. Because a thruster was not selected at this stage of analysis, it was favorable to have the lowest required propellant mass possible for maneuvering to the Science Leg, as extra propellant mass would be required for stationkeeping, as well. Considering initial lifetime estimates and propellant mass needed, it was decided that an orbit closer to 250 km by 400 km would be best for MagSTARS. However, as described in Section 2.2, lower orbits allow for lower altitudes in the ionosphere to be sampled. Therefore, it was decided that a Science Leg orbit that stayed within a set band of altitude would be best, as to keep the lifetime high, the required propellant mass low, and collected a desirable amount of data. The range that was chosen based off the orbits analyzed above was a perigee of 200-250 km and an apogee of 350-400 km. As described in Section 4.3.2, MagSTARS would arrive to its Science Leg orbit at the upper bounds of these ranges, drag would cause it to slowly reach a lower bound, and then a stationkeeping maneuver would be used to raise the satellite back up to the upper bound altitude. This orbit seemed to be well within the performance envelope of the propulsion system described in Section 4.2.

4.3.2 Mission Set-Up

Using STK, a satellite was inserted into a scenario using the Astrogator propagator. In the Mission Control Sequence, MagSTARS' initial state was defined with the Keplerian elements of the initial Sun-synchronous 625 km orbit. 625 km was the highest altitude the launch vehicle would deploy the satellite, so this was chosen as the starting altitude in order to analyze the furthest change in altitude. Analyzing at this furthest point accounted for the maximum required ΔV to ensure enough propellant was available. The initial right ascension of the ascending node (RAAN)

was determined by running the full Mission Control Sequence once it was set up and ensuring that there was almost constant solar intensity during the start of the mission.

The initial spacecraft parameters are shown in Figure 26. The dry mass was estimated based off the subsystems' upper estimates for their components and the average dry mass of the BET-1mN thruster with three tanks, described in Section 4.2. The Solar Radiation Pressure and Radiation Pressure areas were estimated to be 0.0800568 m^2 , as the face towards the Sun is approximately 2U by 3U. The drag area was calculated by averaging the areas of the tumbling faces (approximately 2U by 1U and 3U by 1U) [55]. The area for each face was described in Section 3.3. The drag coefficient was based on estimates for a similarly sized spacecraft cross section at relevant altitudes [55]. The remaining coefficients and K1 and K2, the GPS solar radiation pressures, were left as the STK default values, as these were assumed to be fair estimates for satellites. The fuel mass was set to 0.258 kg, as described in Section 4.2.

	Dry Mass:	7.4432 kg
Drag	Coefficient (Cd):	2.2
	Area:	0.028538 m ²
	Solar Radiation Pressure (Spherical)	
	Coefficient (Cr):	1
	Area:	0.0800568 m ²
Radiation Pressure (Albedo/Thermal)		
	Coefficient (Ck):	1
	Area:	0.0800568 m ²
GPS Solar Radiation Pressure		
	K1:	1
	K2:	1

Figure 26: Initial state estimates for MagSTARS in Astrogator

Because inclination changes were small, it was decided that it was not necessary to perform an inclination change maneuver, as the satellite would still be mostly Sun-synchronous without it and the maneuver would increase the overall requirement for the ΔV budget. Additionally, due to analysis described in Section 4.3.1, it was decided that the Science Leg orbit should not be Sun-synchronous.

After the initial state was defined, a finite maneuver using the BET-1mN was set for MagSTARS to perform a continuous burn against the velocity vector, in order to lower the satellite rather than raise it, until its apogee reached 400 km, as the goal was to have the Science Leg of the mission in orbit with a 200-250 km perigee altitude band and an apogee within a 350-400 km altitude band. This finite maneuver resulted in a perigee altitude of 376 km. To lower the perigee in such a way that would not allow the finite burn maneuvers to lower the apogee, an automatic sequence (autosequence) was added. This autosequence consisted of a finite maneuver which against the velocity vector for 1400 s, or 23.33 min (approximately one-fourth of the period of the Science Leg orbit). This autosequence was added to an apogee stopping condition in a Propagate segment so that each time MagSTARS hit its orbit's apogee, it would run the autosequence to perform a finite burn for one-fourth of its next orbit. The autosequence finite maneuvers would stop occurring when the perigee reached an altitude of less than 250 km by adding this as a constraint to a perigee stopping condition in the same Propagate segment. MagSTARS then began the Science Leg by propagating around the Earth until the apogee dropped to approximately 350 km. This decay would happen due to atmospheric drag, as discussed in Section 8.1.3.1. Thus, orbit raising to stay within the Science Leg orbit altitude range, or stationkeeping, was necessary. To stationkeep, each time the apogee reached 350 km, an autosequence similar to the one previously described was triggered. This autosequence finite maneuver went along the velocity vector rather

than against, but still had a 1400 s duration. It was triggered each time the orbit hit the perigee until the apogee was above 400 km. Based on propellant mass estimations, MagSTARS only had enough fuel to raise the apogee three times using this technique.

4.3.3 Orbital Results

Using the initial set up described in Section 4.3.2, key parameters for each maneuver were found, including ΔV , propellant mass, and burn time. For the maneuvers where a burn occurred each time apogee or perigee was reached, the total time for the maneuver was also recorded. These values are shown in Table 12. The resulting mission lifetime in the Science Leg orbit was 120 days after taking 35.5 days to arrive in this orbit from deployment. 30.5 days of the time in the Science Leg orbit were used for station-keeping maneuvers.

Table 12: Maneuver Summary

Maneuver	ΔV (m/s)	Propellant Mass Used (kg)	Burn Time (min)	Total Maneuver Time (days)
Lower Apogee to 400 km	124.988	0.122	2.2736e+04	16
Lower Perigee to 250 km	40.315	0.039	23.33 per burn	19.5
Raise Apogee to 400 km 1	20.952	0.020	23.33 per burn	10
Raise Apogee to 400 km 2	22.968	0.022	23.33 per burn	11
Raise Apogee to 400 km 3	19.366	0.018	23.33 per burn	9.5
Total	228.589	0.221	4.1310e+04 (~29 days)	66

A final mission lifetime maneuver summary is shown below in Table 13. The colors associated with each stage correspond to the orbit colors shown in Figure 27, which shows MagSTARS' orbit path during different period of its lifetime. Figure 28 shows the apogee and perigee of MagSTARS during its mission. In this figure, the x-axis is the time of the scenario in UTCG from October 29, 2020 to April 12, 2021, and the y-axis ranges from 200 to 660 km and shows the apogee altitude in green and the perigee altitude in black.

Table 13: MagSTARS' Mission Timeline

Date (UTCG)	Apogee Altitude (km)	Perigee Altitude (km)
<i>Launch Vehicle Deployment</i>		
October 29, 2020	625	625
<i>Orbit Lowered (Green)</i>		
November 14, 2020	400	377
<i>Perigee Lowered (Pink)</i>		
December 4, 2020	375	250
<i>Coast Period (Orange)</i>		
December 21, 2020	350	230
<i>Apogee Raised (Purple)</i>		
Dec 31, 2020	400	240
<i>Coast Period (Blue)</i>		
January 25, 2021	350	250
<i>Apogee Raised (Red)</i>		
February 5, 2021	400	250
<i>Coast Period (White)</i>		
March 1, 2021	350	250
<i>Apogee Raised (Army Green)</i>		
March 10, 2021	400	250
<i>Coast Period (Yellow)</i>		
April 12, 2021	350	225
<i>Natural De-Orbit Out of Science Leg Orbit</i>		

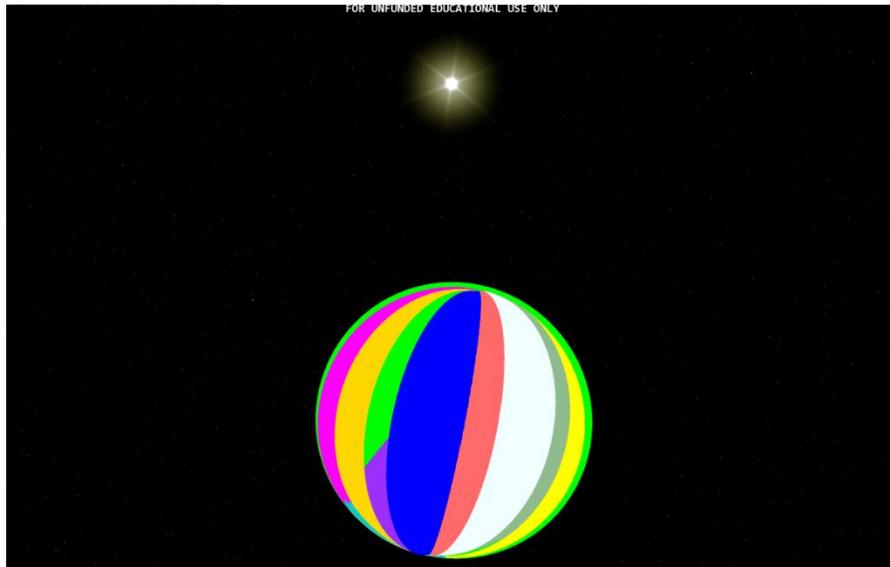


Figure 27: MagSTARS' orbit paths

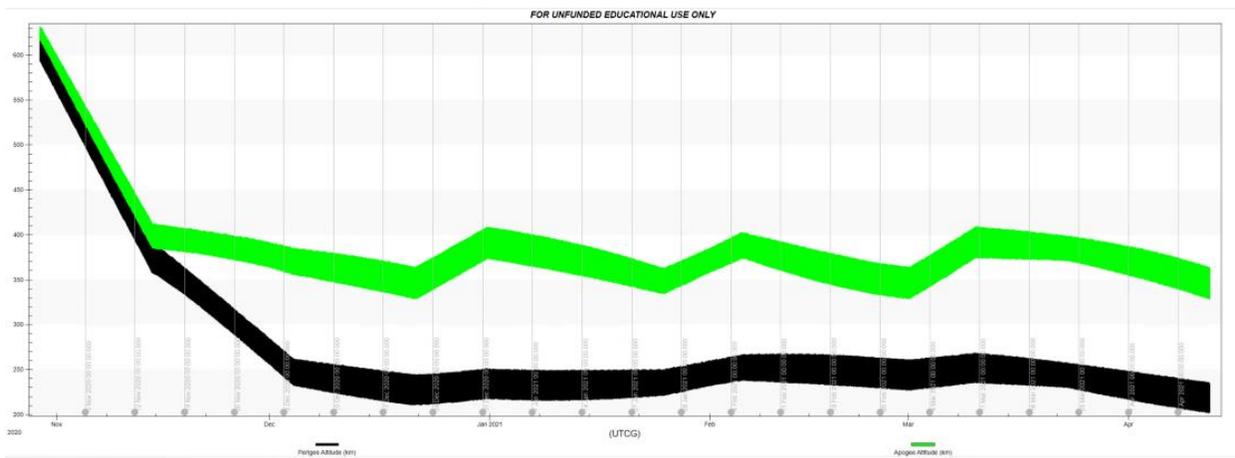


Figure 28: Apogee and perigee throughout mission

As shown by Table 13, stationkeeping extended MagSTARS' lifetime by nearly three and a half months (approximately 103 days). If the apogee was never raised after reaching 350 km on December 21, it would continue to de-orbit after only spending 17 days in the Science Leg orbit.

This would be the case if the propulsion system were to be damaged during MagSTARS' mission to the Science Leg orbit.

As described in Section 4.2, MagSTARS had 0.258 kg of propellant available, but as shown by the above results, only 0.221 kg of propellant were used. The remainder of the 0.037 kg of propellant was saved for any worst-case scenarios. Using the initial STK set-up, three worst-case scenarios were analyzed: MagSTARS being deployed further than the launch vehicle predicted, the thruster not producing as much thrust as anticipated, and a failed stationkeeping maneuver.

MagSTARS' orbital analysis was done assuming that the launch vehicle would deploy the satellite at its upper range altitude of 625 km in order to analyze the predicted worst case. However, in order to add a higher margin of error, higher deployment altitudes were analyzed as well. It was determined that the highest MagSTARS could be deployed in order to still have enough propellant to get to the Science Leg orbit and perform its planned stationkeeping burns described in Section 4.3.2 was 695 km. From this altitude, the ΔV for the maneuver to get to the Science Leg orbit would increase by 38.547 m/s and the burn time for the maneuver would increase by approximately 4.6 days. In another worst-case scenario, if the thruster was only running at 85.5% efficiency for the entire planned mission, the remainder of the available propellant would be needed to make up for the loss.

With limited space in the satellite and a low likelihood of all worst-case scenarios requiring use of the extra propellant occurring during MagSTARS' mission, combinations of lower magnitude errors were tested. On average, based on the values in Table 12, the propellant needed to perform one apogee-raising, stationkeeping burn was 0.020 kg, the average ΔV was 21 m/s, and the average time to complete the full apogee-raising maneuver was approximately 10.2 days. Needing to perform an extra full stationkeeping burn (raising apogee back to its upper bound on

400 km) would leave a remaining 0.017 kg of propellant for error. This would be enough to either be deployed by the launch vehicle at an altitude as far as 648 km or for the thruster to only be running at 93% efficiency. Being deployed at 648 km would increase the total ΔV by 12.935 m/s and would increase the burn time and arrival to the Science Leg orbit by approximately 1.5 days.

A final case considered was the case that MagSTARS was deployed further than anticipated and the thruster was not performing at top efficiency. One example of this that was analyzed was if each error used half of the remaining propellant (0.0185 kg each). This would allow the thruster to run at an efficiency as low as 92.3% and would account for a deployment altitude as high as 660 km. This high of a deployment altitude would increase the overall ΔV by 19.444 m/s and would delay arrival to the Science Leg orbit by slightly over 2 days.

These are just a few example cases of different scenarios that could occur. Many other combinations of deployment altitude, thruster efficiency, and stationkeeping success exist, as well. With 0.037 kg of remaining propellant, it is highly likely that MagSTARS will have enough propellant to account for the most likely fuel mass-dependent errors that may occur and still be able to complete its planned mission.

5 Power System Design and Analysis

This chapter describes the power subsystem of MagSTARS including the component selection process and power system analysis. Both STK and MATLAB were used to create a model of satellite power generation, use and storage that was utilized in satellite power analysis.

5.1 Power Overview

The power subsystem fulfilled three primary functions: the collection, storage, management, and distribution of electrical power. Firstly, power collection on CubeSats is done using solar cells mounted to panels in various configurations depending on the satellite. In this case, they will be body mounted as MagSTARS will be continuously spinning for much of the mission, removing any distal or deployable solar panel system from consideration. Extended panels would increase the moment of inertia of MagSTARS as well as creating non-symmetrical geometry, leading to slower spin up and spin down, and a more complicated control scheme.

CubeSats use highly efficient gallium arsenide (GaAs) solar cells, with beginning of life efficiencies of approximately 30% and end of life efficiencies of approximately 25% [56]. Available prefabricated panels generally are capable of 2.3 to 2.8 W per 1U area, and it is expected that this project's 6U CubeSat will continuously generate approximately 15-19 W once in orbit based on commercially available 6U panels using GaAs solar cells [34], [35], [36].

Solar cells are accompanied by supporting electronics in the form of a power conditioning unit (PCU) that help manage the solar cells. This means controlling solar panel power output, as well as ensuring continuing functioning of solar cells even in the case of a cell failure. By adjusting the current draw from the solar cells, the PCU can maximize solar cell power output, as shown in Figure 29 below [56]. This is known as maximum power point tracking (MPPT) and is used on

CubeSats to ensure solar cells capture the most power possible. Due to the nature of solar cells, as current draw increases, the voltage decreases, as shown by the blue curve in Figure 29. Due to the relation of electrical power to voltage and current,

$$P_{electrical} = I \times V \quad (5)$$

the optimal power output occurs in the region where the product of current and voltage are the highest. This is shown on the red power curve between point 2 and 1. Additionally, the PCU can directly charge the batteries from the solar cells at reduced efficiency if MPPT electronics fail. By installing bypass diodes, sets of cells in series can continue supplying power at a reduced voltage when one or more cells do fail, allowing for functional cells to continue power generation [57].

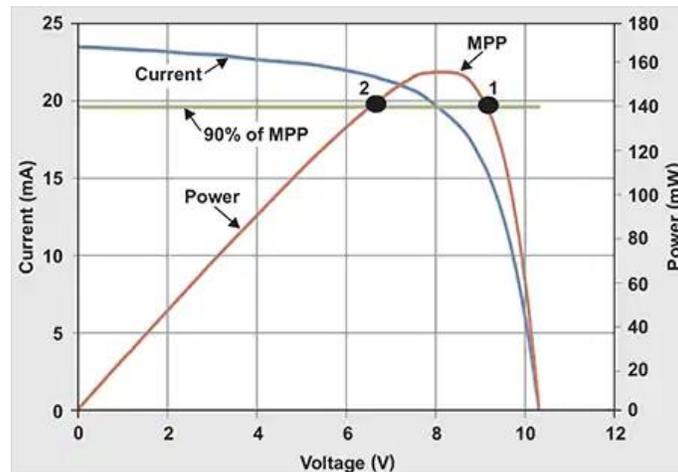


Figure 29: Graph of solar cell power output compared to solar cell current and voltage [56]

Power storage on CubeSats is done using batteries, generally lithium ion, lithium polymer, or lithium iron phosphate (LiFePo₄) type batteries. These batteries provide a high energy density of approximately 100–250 Wh/kg, making them ideal for power storage on small satellites [58]. However, these types of batteries only remain functional within a certain temperature range, operating at optimal efficiency between 15–35 °C, lower levels of performance between -20-60°C,

and at risk of cell damage or failure within -40 – 70°C [59]. Due to extreme temperature variation experienced by satellites during the transition from being fully lit by the Sun to the shade of the Earth's penumbra, in the order of hundreds of degrees Celsius, it is important to manage battery temperature to maintain optimal battery performance. This is done using heaters built into the battery system, or various heat rejection methods discussed in Section 9 to keep batteries within their optimal temperature range when in shade or Sunlight respectively [34], [35].

Finally, power distribution and management involve providing several sources of power at varying voltages, as well as monitoring each power bus for current draw or potential issues. Generally, satellite power management/distributions units (PMU/PDU) use voltage regulators to supply a target voltage to each power bus as needed by connected subsystems. Each bus is protected by a switch controlled by the PMU as well as monitored for current draw, to allow for the tracking of power use by each power bus [60]. An example model of this system is shown in Figure 30 below.

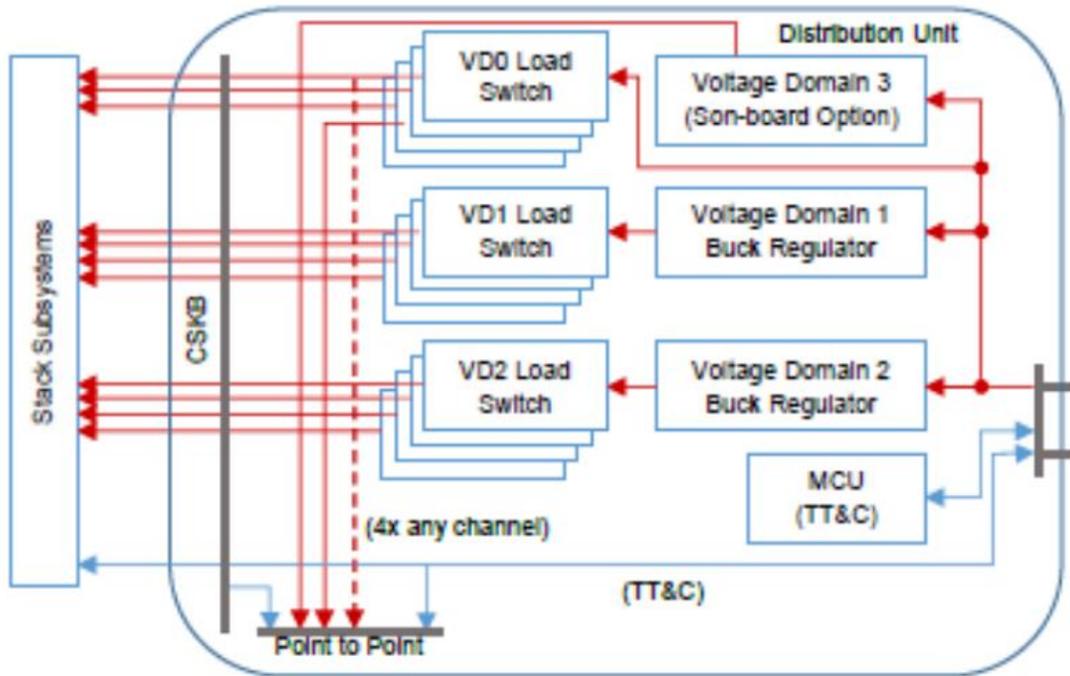


Figure 30: ISISpace PDU system diagram [36]

When selecting the components that will make up the power subsystem, it is first important to understand system requirements and constraints. Thus, the objectives set forth in Section 1.3 for the power subsystem were followed during the development of the power subsystem.

5.2 Component Selection

As the project approached completion, and mission characteristics are better defined, the accompanying analysis and selection for the power subsystem became more detailed. Initial analysis for the power system used more generalized approximations, while analysis for the final system used software and more comprehensive modelling to verify power requirements were met. The analysis done focuses mainly on the power subsystem components, as well as the subsystems with the highest power requirements: propulsion, ADCS, and communication.

To select power subsystem components, constraints were identified based on known mission parameters to act as the basic requirements for the system. Starting with physical constraints, due to the rotation of the of the CubeSat it was determined that body mounted solar panels were required, as any folding system would cause complications for the ADCS. In addition, power subsystem volume and mass were to be low as possible. In terms of important electrical properties, the power system was to have highly efficient power collection, battery charge/discharge and regulation/conversion systems. This system needed to be highly reliable, flexible in terms of voltages supplied, and capable of high-power outputs of at least 15 W during peak power usage.

During the component selection process, it was considered desirable to use components from the same manufacturer, as most available components are part of an integrated power system offered by commercial providers. By buying a complete system from a single manufacturer, it is reasonable to expect all power system subcomponents to be fully compatible. This is not always the case when using various components from different providers, as each manufacturer may use different bus voltages, electrical connections, physical specifications, and communication protocols. Therefore, chosen components will be part of one manufacturer's full power subsystem.

The power system was selected using a decision matrix approach, to weigh the pros and cons of each and determine the best choice for fulfilling mission parameters. For each system different factors were considered, based on that components function. These factors include component physical properties, such as mass, volume, form factor, and available interfaces. In addition to these basic constraints, functional features of each system were also compared. Features such as solar cell efficiency, battery depth of discharge (DOD), battery capacity, battery charge/discharge rates, and idle and operational power use.

Table 14: Power Subsystem Decision Matrix

Power Subsystem	Mass	Space	Batt Capacity	Flexibility	Documentation	Bat Max Pout	Total Score
ISISpace MEPS	2	1	2	2	2	2	11
Clyde Space MPS	1	2	1	1	1	1	7

Based on the results of this matrix, ISISpace’s modular electric power system (MEPS) was selected, along with ISISpace solar panels. Due to the high technology readiness level (TRL) and long flight heritage of all ISISpace hardware, their equipment provided a reliable choice. This system is composed of five major components, the Solar Panels, Power Distribution Unit (PDU), Power Conditioning Unit (PCU), Power Battery Unit (PBU), and the Power Battery Pack (PBP). Each manages different responsibilities of the power subsystem: power generation, power distribution and management, power management and power storage, respectively. This system is shown in the diagram below.

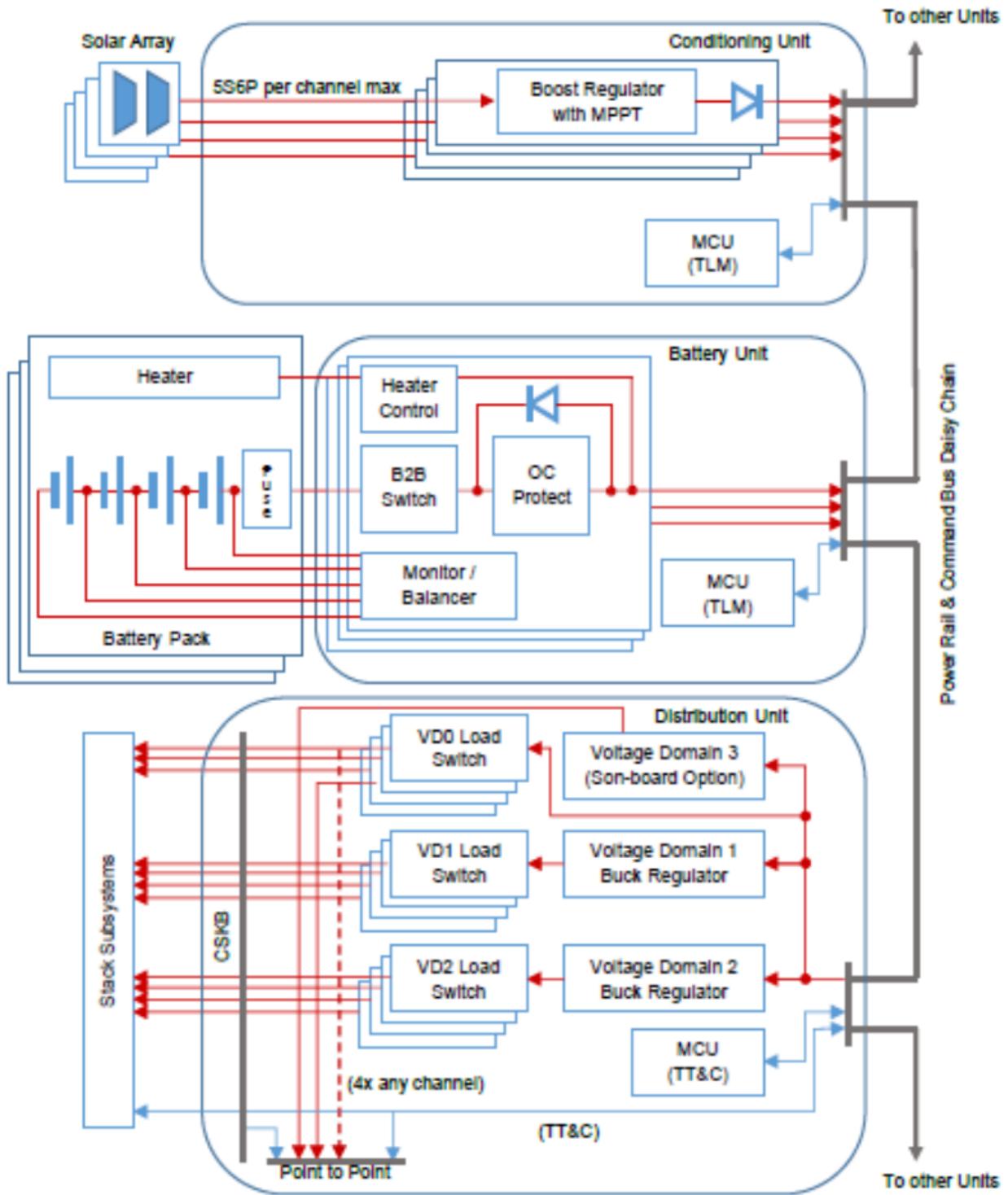


Figure 31: ISISpace MEPS system diagram

The ISISpace MEPS modules are designed to be stacked as shown in Figure 32, allowing for simple power system integration with a relatively compact form factor. The PDU utilizes the PC-104 connector bus, a common stackable connection system for CubeSat electronics for ease of connection to additional “breakout board” and subsystems, however the connections between MEPS modules are made using jumper wires with Harwin M80 L-Tek connectors in a daisy chain configuration. While MagSTARS does not require battery capacity in excess of a single battery module, it is possible to have several of each module in a stack to provide additional capability to the power system.



Figure 32: Fully assembled ISISpace MEPS stack

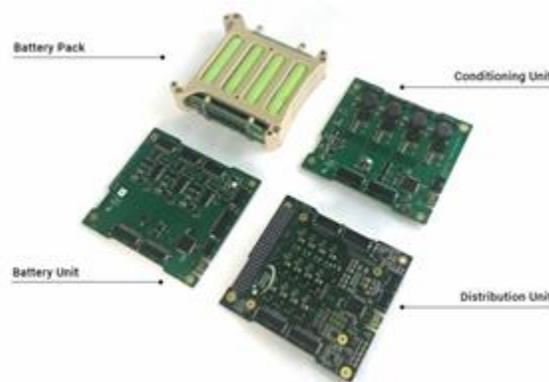


Figure 33: Four primary MEPS modules (PBP, PCU, PBU, PDU)

The solar panels selected for MagSTARS are a pair of 1x3U panels, a 2x6U panel and a 1x2U panel provided by ISISpace [35], rated for generating 6.9 and 17 W, respectively. These panels utilize gallium arsenide (GaAs) solar cells manufactured by AZUR Space, with a beginning of life (BOL) collection efficiency of 29.5% at 28°C [61]. However, the steady state temperature of MagSTARS while in space is approximately 350 K (76.85°C), which reduces the collection efficiency to approximately 28%. The 1x3U panels are mounted to the corresponding 1x3U sides of MagSTARS, while the 2x3U and 1x2U panels are mounted on MagSTARS' 2x3U sides. A 1x2U panel is used on one 2x3U face to leave room for the patch antennas utilized by the telecommunications system. This leaves the 1x2U faces for use by the propulsion system and mini-INMS. These panels are in a similar configuration to that of the panels on the ISISpace 6U CubeSat platform pictured below in Figure 34.



Figure 34: ISISpace 6U CubeSat platform

MagSTARS' solar panels were connected to the PCU, with each panel connected to one of the PCU's four solar array input channels. Solar panel power output was managed by the PCU, which utilized max power point tracking (MPPT) to ensure solar cells output the most power possible in all conditions. The PCU achieves this by adjusting the current draw from the cells by

its regulator units as to optimize the amount of power output by the panels. This maximizes the overall power collected by the panels, even in varying thermal and lighting environments as seen later in MagSTARS' mission. The regulators used by the PCU are boost converters, which are more than 90% efficient, even for extremely low loads of 5 to 10 mA [62]. This means that most of the power generated by the solar panels can be utilized by other power boards or subsystems, and less than 10% of collected power is used in the conversion process or lost as heat. The boost converters output to the main power rail at 12.8 – 16 V for use by other power boards.

The PBU controls the charging and discharging of the PBP based on satellite power requirements, using available rail power. When all other subsystems require less power than the solar panels are producing, the PBU draws rail power, charging the PBP until it is full. When more power is required than supplied by the panels, the PBU outputs power from the PBP to the main power rail. Based on testing done by ISISpace on this power system, the entire chain from solar panel input to battery output should be anywhere from 80 to 94% efficient depending on current draw from the battery at the expected rail voltage of 12.8 to 16 V [35].

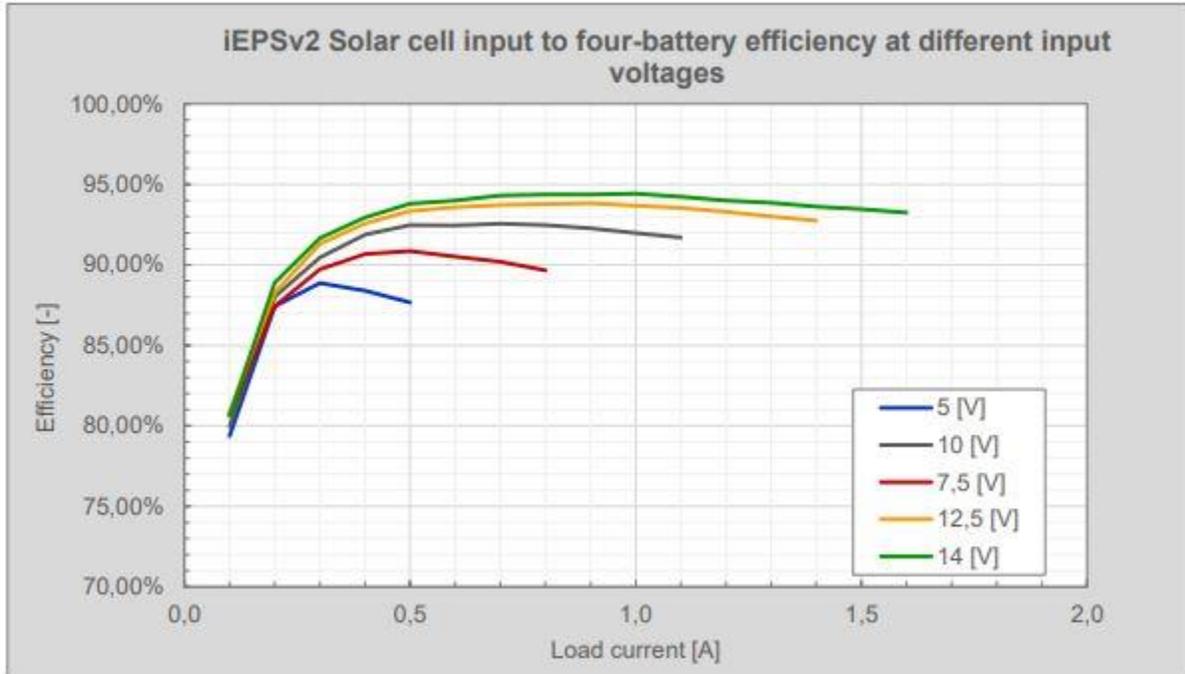


Figure 35: Solar cell – battery system overall efficiency at different input voltages [35]

The power distribution unit, (PDU) converts rail power down to lower voltages used by satellite subsystems, as well as monitoring these outputs and providing various electrical protections for the system. There are three output buses, one at rail voltage (12.8 – 16 V), one at 5 V and one at 3.3 V, each containing 4 independent channels. Each bus is supplied power via buck converters, with a current limit of 3 A per channel, and a 4 A continuous/6 A peak per bus. All output channels, overall bus output, and overall power input are monitored for voltage, current and power. In addition to circuit monitoring ISISpace’s PDU has extensive protections in place for its outputs as well as its inputs, with every output/input channel provided overvoltage, overcurrent, and thermal, and reverse current protection by the PDU. In the case of extremely low available power, the PDU can operate in an emergency low power mode to conserve energy.

5.3 Solar Array Power Generation Modeling

STK was used to model MagSTARS' solar power collection and access use during different mission phases to obtain power supplied by the solar panels, the time periods when transmitting which were used to calculate transmitter power use, and the periods for which the propulsion subsystem was active. Calculating power generation and use during each mission phase was important as it provided total power available for use by all subsystems. It was also used to create detailed power budgets for each mission phase and schedule subsystem power use if necessary. This was done using the orbital scenario created by the propulsion subsystem described in Section 4.3.1, with analysis performed for each mission phase from CubeSat launch to 120 days into the science orbit.

5.3.1 Solar Power Analysis

To model MagSTARS' power generation it was necessary to properly set up a 3D model for use in STK. The team used a mix of manufacturer supplied CAD models and custom models made in SolidWorks to create this model, based on a 6U frame with two 2x6U and two 1x3U body mounted solar panels, as shown below in Figure 36:

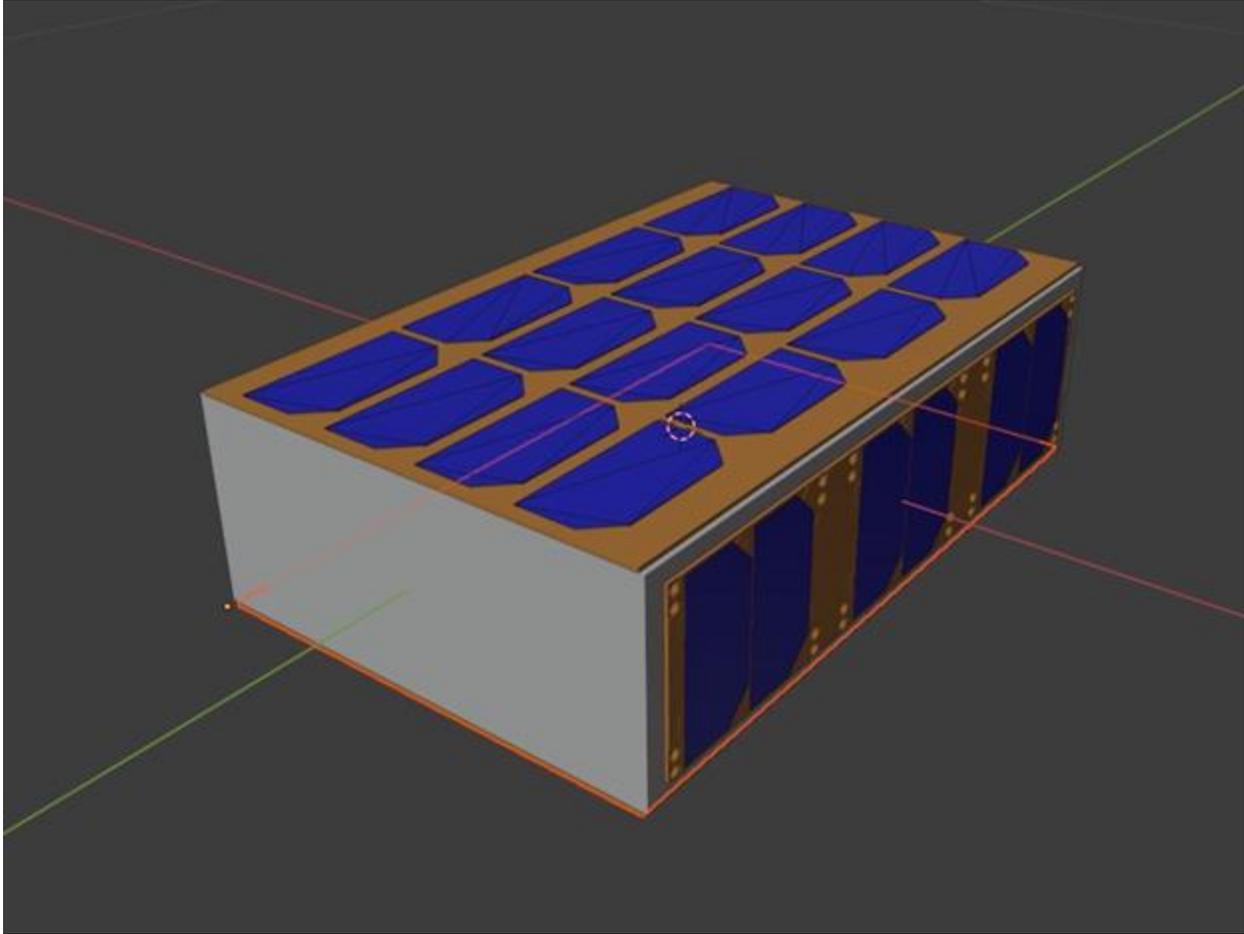


Figure 36: Model of MagSTARS in Blender.

Using the AGI provided guide on creating a 3D satellite model for solar power modelling [63], the previously created 3D assembly from SolidWorks was imported to Blender as an .STL file, and formatted for use in STK. For further information on the setup of this 3D model please reference AGI's documentation.

Using the STK scenario provided by the propulsion subsystem, MagSTARS' 3D model was set to the previously formatted model. Using the Solar Panel tool, the CubeSat's power generation was plotted for all mission phases for a half day with a sampling rate of 60 seconds. STK's solar panel tool uses Eq. (6) to calculate the power output per panel [64], [6];

$$P_{sa} = \eta_{sa} A_{sa} G \cos(\theta) I \quad (6)$$

where P_{sa} is the solar power output, η_{sa} is the solar array collection efficiency, A_{sa} is the solar array area, G is solar flux, θ is solar incidence angle, and I is solar irradiance.

As stated above, this equation accounts for solar cell efficiency, the solar flux based on the satellites distance from the Sun, the effective area of the array using solar incidence angle and array area, and the solar irradiance. This analysis used a fixed solar panel collection efficiency of 28%, chosen based on an estimate of the steady state temperature of the satellite, 350 K, and panel degradation after exposure to $2.5 \cdot 10^{14}$ electrons at 1 MeV [64]. Solar array effective area is based on the area of each array and that array's inclination angle to the Sun and is represented by θ . This is the area normal to the Sun vector.

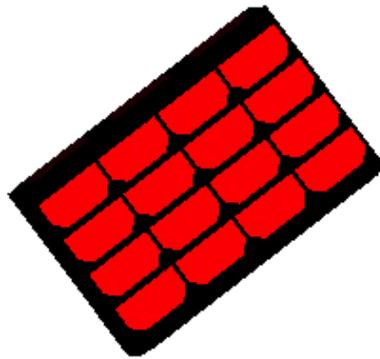


Figure 37: Solar cell illumination in STK [64]

Finally, the solar irradiance provides the intensity of the Sunlight on the solar array, from 0 to 1, based on obstructing objects such as the satellite itself, Earth, and the atmosphere. Overall, this analysis provides a very good estimate for the power generated by each solar array, which, when combined, provided the total power generated by all solar arrays on the CubeSat.

5.3.2 Solar Power Results

Utilizing the STK Solar Panel Tool, a trade study was performed over the span of MagSTARS' mission to better understand the power generation profile and how it will change over time. This was done by running analysis every 30 days for 24-hour periods up to 120 days into the science phase. This yielded a power generation profile plotted over the span of the mission, and MagSTARS' expected power generation would change over time as shown in Figure 38 below. Subsequently, this process was repeated several times for various orbits later considered, to provide information on the long-term power generation profile provided by each orbit. Of the orbits considered, the orbit with a 250 km perigee and 400 km apogee provided the most favorable power generation profile, with average power generated staying above 15 W for the first approximately 165 days of the Science Leg. This orbit, shown below in Figure 38, was selected by the team for MagSTARS.

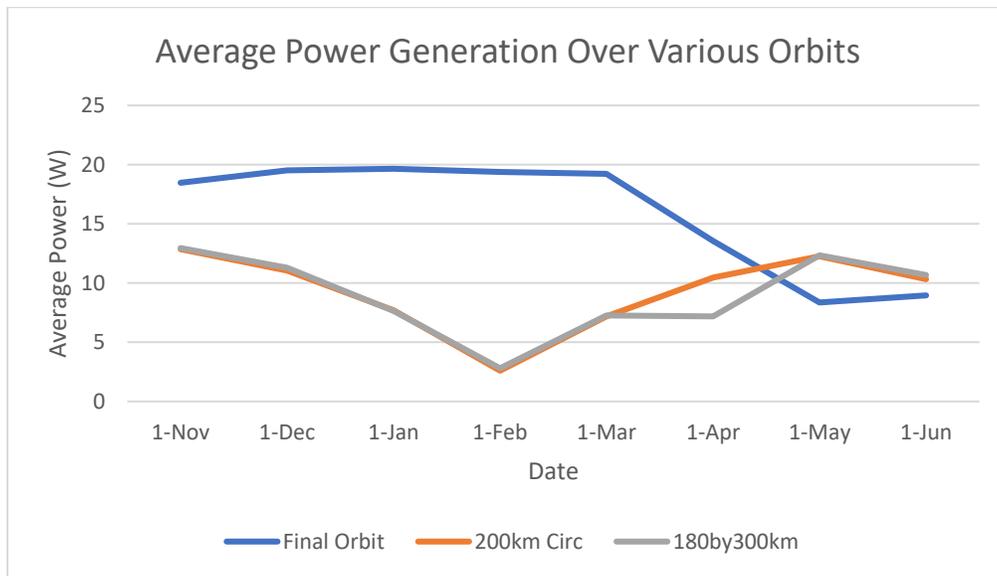


Figure 38: Plot of average power generation over mission for different orbits

The power generation profile is critical when considering the feasibility of the chosen orbit, power system and science mission as the spacecraft design progresses. The power profile shows MagSTARS remains in optimal conditions and generates approximately 19 W continuously for approximately 120 days in the selected orbit, while in other considered orbits, average power generation drops off significantly earlier. After this time, power generation begins to drop off as the solar inclination angle and the period spent in eclipse increase.

During this later period, it becomes necessary to schedule power for subsystems with high power requirements, such as propulsion or comms. Battery power output and capacity also become relevant due to the increasingly prolonged periods of eclipse, for which the battery must supply power. As solar power is no longer supplied continuously, it is important to study the solar power output in more detail. The difference between the power generation during before 120 days into the science phase and after is shown below in Figure 39 below. This data was generated with a 60 second sample rate and provides a relatively high-resolution plot of solar power over the 24-hour analysis period. These plots show the power generation of each solar array on MagSTARS and overall power generation of all arrays combined (yellow). As you can see, the continuity of power generation changes greatly after 120 days leading to cyclical period of power generation every orbit. Up to and including 120 days into the science orbit, the CubeSat is in constant Sunlight, and hence afforded constant power generation. Also, at this point, the primary 6U solar array is still pointed almost directly at the Sun, maximizing solar power collected while in Sunlight.

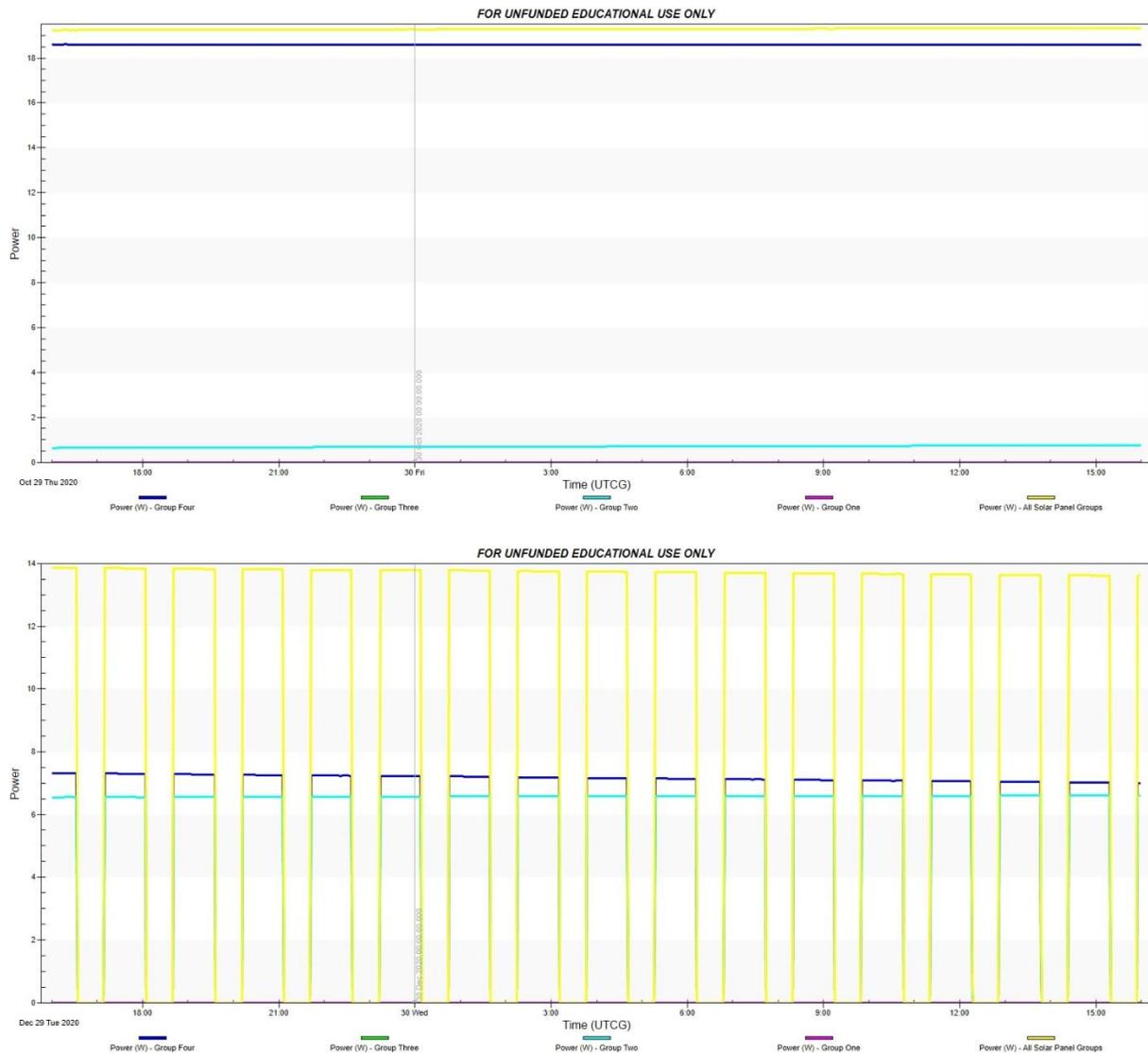


Figure 39: Detailed power profiles before and after 120 days

In addition to modelling power generation of satellite solar panels, STK was used to obtain access times for the CubeSat RF (Radio Frequency) transmitter and the propulsion subsystems operation periods. This data was utilized within the overall power model to determine mission Con-Ops, such as when to transmit data and fire the primary propulsion system. Due to the high-power draw of the transmitter of approximately 5 W, as well as the propulsion system's 5 to 15

W, knowing when and for how long these systems will be active is especially important for the overall power budget and scheduling, and for use in the finalized dynamic power model.

5.4 Dynamic Power Model

After determining the power generation via STK's Solar Panel Tool, further analysis was done using MATLAB. The MATLAB model takes the power generation data calculated by STK, along with power use profiles for all other subsystems to dynamically simulate MagSTARS total power in, storage and output. This allows for different cases to be tested for feasibility with the selected power system.

5.4.1 Dynamic Power Analysis

One subsystem with a significant power requirement is ADCS, specifically the magnetorquers used for MagSTARS' attitude control, with a max power draw of approximately 1.2 W and substantial active periods. Due to the high-power draw of this component, it was desired for magnetorquer power consumption to be dynamically modeled as part of a MATLAB control algorithm created by the ADCS subsystem. In addition to its primary function, controlling CubeSat attitude, this script also calculates the total magnetorquer required power based on current supplied to each coil during adjustments made by the control algorithm.

As previously stated, transmitter access times were obtained using STK. Within the MATLAB model, it is assumed the transmitter operates at max power, drawing about 5 W while on. Based on whether the transmitter has access to ground stations, the state of the transmitter is set to be on when access is available, and off when there are no available accesses.

During mission legs 1-2, MagSTARS' onboard propulsion system will be used to maintain its orbit. Using thruster operational period data obtained in STK, thruster power use is modelled

in a similar manner to that of transmitter power uses, with a predetermined power draw applied during periods of operation.

All other subsystems were assumed to have continuous power draw, at their posted maximum power requirements, as shown in Table 15 below. This table does not account for the power use of the propulsion or telecommunications subsystems.

Table 15: Subcomponent Power Requirements

Component	Science Phase Power Req (W)	Propulsion Phase Power Req (W)
Mag-Torquers	1.2	1.2
GPS	0.165	0.165
Sun Sensors	0.05	0.05
Magnetometer	.0104	0.0104
Control Board	0.4	0.4
Accelerometer	0.15	0.15
Gyroscope	0.3	0.3
Mini-INMS	1.8	0
IPDU	0.066	0.066
PCU	0.066	0.066
IPBU	0.063	0.063
Total Power	5.19	3.39

5.4.2 Dynamic Power Model Results

Once the power model was completed, analysis was run over specific periods of the mission where power availability was of concern, such as propulsive maneuvers and science collection periods. During these periods, based on more general analysis, the CubeSat will be operating close to or at a net loss of power depending on power budgeting. This analysis provides a more detailed profile of satellite power performance over these periods, allowing for power budgeting and operational scheduling of high-power systems that ensure optimal battery depth of discharge and power usage.

First, the propulsive maneuver periods were analyzed, as the propulsion system had the highest power requirement of 15 W. As all propulsive maneuvers take place before 120 days into the science leg, the solar arrays generate approximately 19 W of power continuously. For these periods there was assumed a base power use of 3.39 W for all systems other than propulsion and telecommunications. Telecommunications is assumed to receive once every orbit for a period of 5 minutes, with a power requirement of 4.5 W. As the mini-INMS is not in use, it was assumed that satellite transmission would be negligible, as the telemetry data rate is quite small compared to the instrument data rate, and therefore requires minimal transmission time. Two type of propulsive finite maneuvers are executed throughout the mission, continuous burns, and periodic burns. As shown below in Figure 40 and Figure 41 during propulsion periods there is sufficient power thus MagSTARS does not run a deficit over time and can operate all systems continuously, both in constant and in periodic finite maneuvers. Additionally, once the battery reaches full capacity, the periodic depth of discharge (DOD) is very low, approximately 1% for the constant propulsion case and 0% for the periodic case. This is far above the maximum allowable DOD for the mission length of approximately 40% and ensures minimal battery degradation during the mission [59].

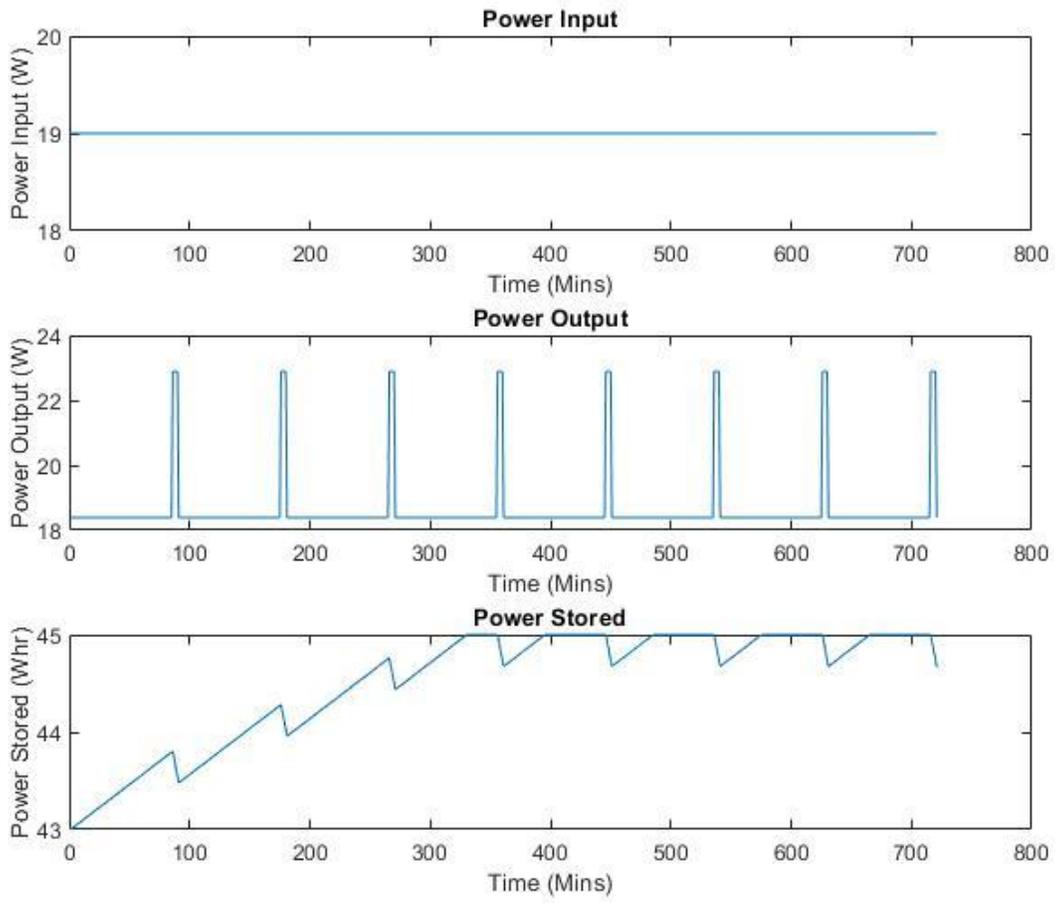


Figure 40: Power use during constant propulsive maneuvers

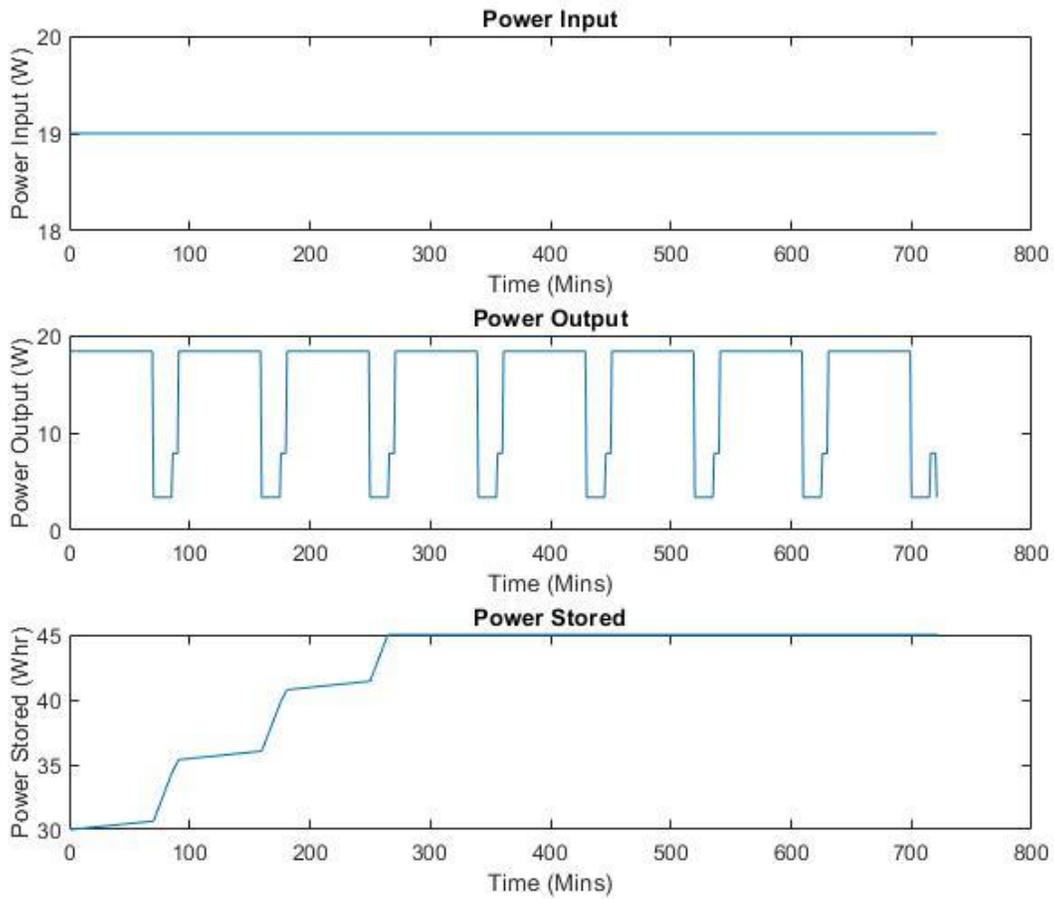


Figure 41: Power use during periodic propulsive maneuvers

During the later period of the Science Phase, power generation becomes restricted due to MagSTARS spending an increasing amount of time in the Earth's penumbra and suffering from an increasing incidence angle between the primary 6U panel and the Sun. Examined in Figure 42 below is the power analysis done at 120 days into the science phase, very close to the end of the mission. Here the average power generated is approximately 13 W, in cycles of shade providing 0 W of power and Sunlight providing approximately 18 W. During this period propulsion is no longer operating and therefore no longer requires power. However, the mini-INMS requires 1.8

W, and creates a substantial amount of data to be transmitted. This requires telecommunications to transmit and receive periodically every orbit. It was assumed there was a 1-minute transmission period with a power requirement of 13 W every orbit, and an overlapping receiving period of 5 min with a power requirement of 4.5 W per approximately 90-minute orbit. This 4.5 W included the 13 W power requirement of the transmitting portion. This provides a significant buffer to transmit all data, as all science data collected can be transmitted in a span of 4 seconds. As shown in Figure 42, the power generated 120 days into the science leg is sufficient, and MagSTARS does not run a long-term power deficit. Additionally, once the battery reaches full capacity, the periodic DOD is approximately 7%, once again significantly less than the max allowable DOD of 40%. This means at 120 days into the science leg, shortly before the deorbiting of MagSTARS, there is still sufficient power to operate MagSTARS in data-collection mode.

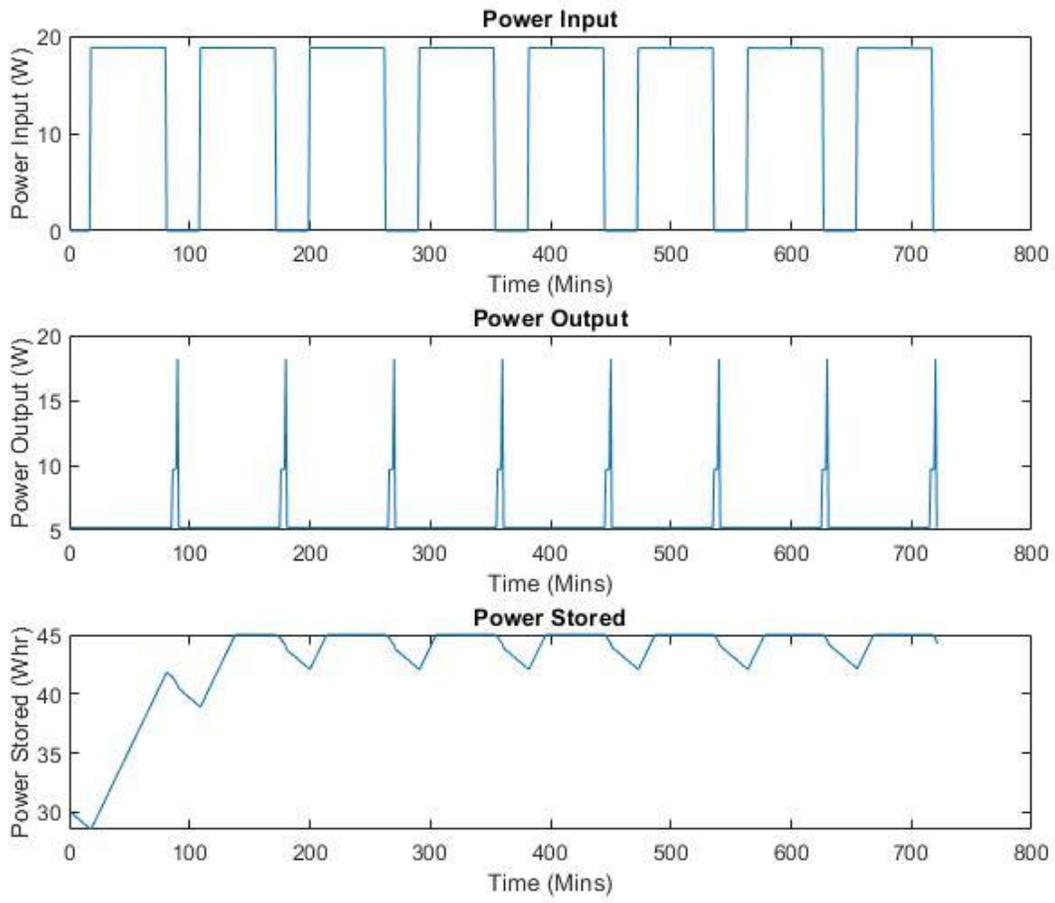


Figure 42: Power use near the end of the Science Leg

6 Telecommunications Design

This chapter discusses the design and analysis of the telecommunications subsystem. Component selection, ground station setup, and an access report analysis will be presented in this chapter.

6.1 Telecommunications Overview

The telecommunications system in a CubeSat is responsible for receiving and transmitting data between the spacecraft and ground stations. Data includes spacecraft housekeeping information and telemetry, as well as the science data collected by the payload. The telecommunications system of a CubeSat primarily consists of three main hardware components: a transmitter, a receiver, and an antenna. These work together to upload and download data to and from ground stations on Earth [37].

Often the transmitter and receiver devices are combined and commonly referred to as the transceiver, as is the case for this project. Separately, the role of a transmitter is to transmit the data from the CubeSat to the ground stations on Earth (downlink), while the receiver receives data sent from the ground station to the satellite (uplink). For the duration of the mission, data from the science payload was stored on one of two 32 GB SD cards in the computer onboard the satellite until ready to be transmitted. While the transceiver connects to and work with the computer on the CubeSat, the antenna is the physical component responsible for the uplink and downlink of the data. Antennas need to operate on the same frequency as the transceiver and receiver. The transceiver takes this data and sends it along to the antenna that generates the radio waves to be transmitted to the ground station. Similarly, the antenna on the CubeSat also receives the information in the form of radio waves that are sent from the ground stations and converts it back

into data that the receiver on the satellite can read. The telecommunications hardware for a CubeSat is dependent on the science mission, the orbit, and the altitude. These parameters translate into the signal strength and coverage required by the telecommunications hardware along with the power consumption for the equipment's "on-time" duration [38], [37].

As mentioned, ground stations are where the communications and data from the satellite are sent to and from in the form of uplinks and downlinks. Ground stations contain two main components, including mission control clients (MCC) and authentication servers (AUS). A MCC is the basic control station for the satellite in orbit and will send commands to the satellite. The AUS is responsible for the communication between the ground stations and the MCC [65].

Institutes such as NASA have created networks of professional ground stations around the globe that are available to commercial and private entities. These networks allow for a larger and longer access time to the satellite. Examples of these networks include the Global Educational Network for Satellite Operations (GENSO), which has since been discontinued, and the Near-Earth Network (NEN) [66]. Amateur ground stations have also been developed for research and amateur missions. Many of these amateur stations can be found at universities, including in the United States such as at California Polytechnic State University, University of Texas, University of Kentucky, and University of Michigan and are a primary mode of CubeSat mission communication for students and researchers [67].

6.2 Component Selection

During the selection of the telecommunication system the team investigated the LEO satellite requirements along with data requirements based off other subsystems. The team first considered radio frequencies, followed by ground station compatibility, and finally data

requirements. In Section 1.5.1.6 radio frequencies suitable for LEO including VHF, UHF, and S-Band were discussed. Table 16 below shows the range for each of these 3 frequencies [2].

Table 16: Radio Frequencies [2]

VHF	30 MHz – 300 MHz
UHF	300 MHz – 3 GHz
S-Band	2 GHz – 4 GHz

As MagSTARS is operating in LEO, any of the three radio frequency bands would suffice. To further narrow down the radio frequencies, the team next considered the Amazon Web Service (AWS) Ground Station network. This ground station selection is discussed in more detail in the following section. The AWS Ground Stations operated in S-Band and X-Band frequencies, narrowing down the team’s selection to operate in S-Band, as the UHF and VHF options would not be compatible [68]. Knowing the team was looking for a high technology readiness level (TRL), research was conducted on common small satellite supplier websites. Ultimately, the team decided on IQ Wireless’ SLink-PHY S-Band Transceiver and Patch Antenna. This pair operates on the lower of the S-Band range at about 2.0-2.3 GHz. The transmission data rate is about 20 Mbps while the receiving rate is about 64 kbps. Both transceiver and patch antenna had a TRL of 9 [69], [70]. The payload’s data rate was about 13.1 kbps and any instructions related to propulsion and ADCS would only be about 10 kb in size which fit into the component's data specifications.

6.3 Ground Station Selection

A key aspect to selecting the ground stations was making sure the locations would provide ample coverage and access to MagSTARS. The team considered three different networks of

ground stations, GENSO, NEN, and AWS, before ultimately deciding on AWS. Each network was researched and weighted to the team's needs.

Initially the team looked at the GENSO network of ground stations, which contained the largest coverage by geographical location. As mentioned previously, it was soon discovered that this operation was no longer active. The NEN was researched and found to still be active with a variety of locations all around the globe. Upon an analysis in STK, it was noted that the access times to the satellite were about 5 minutes for each pass. While working to optimizing these times, AWS Ground Station was discovered. AWS Ground Station is a service that allows users to communicate with their satellites via the ground stations that Amazon already has in place. AWS would also take care of maintenance and use their cloud storage system to store any data received from the satellite [71]. Table 17 below shows eight locations chosen from 24 AWS locations provided [68].

Table 17: Ground Station Locations Provided by AWS

Location
Bahrain, Middle East
Cape Town, South Africa
Columbus, Ohio, US
London, England
Salem, Oregon, US
Sau Paulo, Brazil
Sydney, Australia
Tokyo, Japan

These eight locations were then mapped in an STK scenario from Section 4.3. Figure 43 below illustrates the location and the coverage of each ground station. The eight locations were strategically chosen to span as many regions as possible. The following section will describe the numerical coverage including access times provided by this ground station network.

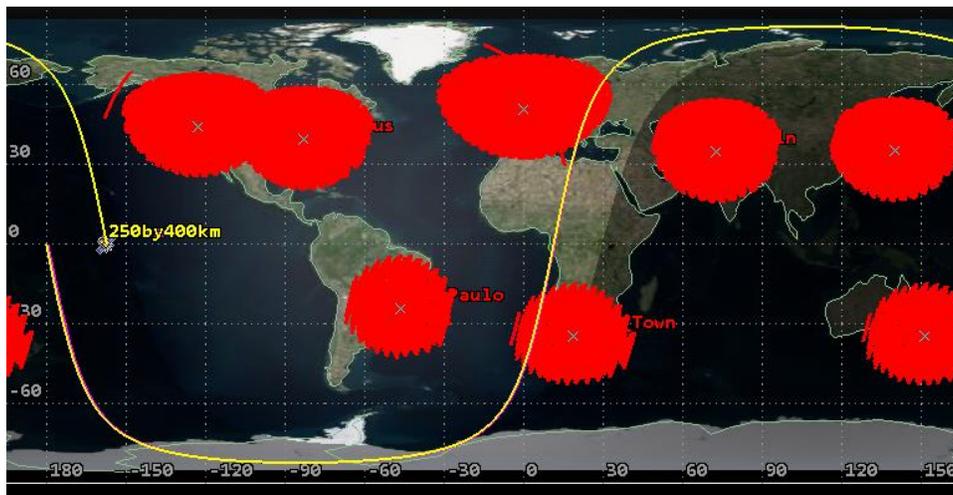


Figure 43: STK ground station coverage map

6.3.1 Recommended WPI Ground Station Set-Up

Part of this project was devoted to researching what would be required to set up a ground station at the WPI. The team ultimately decided that the AWS Ground Station system was the best option with respect to currently available cost and resources. However, going forward, the WPI Aerospace Engineering Department would still like to set up a ground station for future projects and missions. The team researched components that make up a ground station, such as receivers, ground antennae, software, PCs (with Linux operating system), power supplies, and a rack for all the components. While conducting this research the team came across a handful suppliers who sold ground station “kits”. These kits included almost if not everything required to set up a station. This makes gathering all the individual components much easier as they are bundled together and are confirmed to be compatible with each other. It is the team's recommendation to investigate kits such as the ISIS Full Ground Station Kit for VHF/UHF or (S Band) when setting up a ground station at WPI. The cost for these kits runs from about \$75,000 to \$80,000 [72]. The main components included in these kits are as follows:

- Instrumentation Rack containing:
 - S-band ground station receiver
 - Rack mount PC with Local Ground Station (LGS) software
 - Rotator Controller
 - Cavity filters to suppress UMTS interferences. [72]
- Steerable Antenna System
 - Azimuth and elevation rotators with speed up to 60 /sec.
 - Hot-dip galvanized steel mounting mast
 - 3m mesh dish with helix feed, LNA and cavity filters for S-band (2200 – 2290 MHz or 2400 – 2450 MHz)
 - Lightning protection system

- 20m of cable between 19"rack and antenna [72]
- Standard Software
 - Satellite tracking software pre-installed
 - Cavity filters to suppress UMTS interferences.
 - Debian/GNU LINUX operating system pre-installed [72]

A more detailed look at the setup of this ground station can be seen in Figure 44 below.

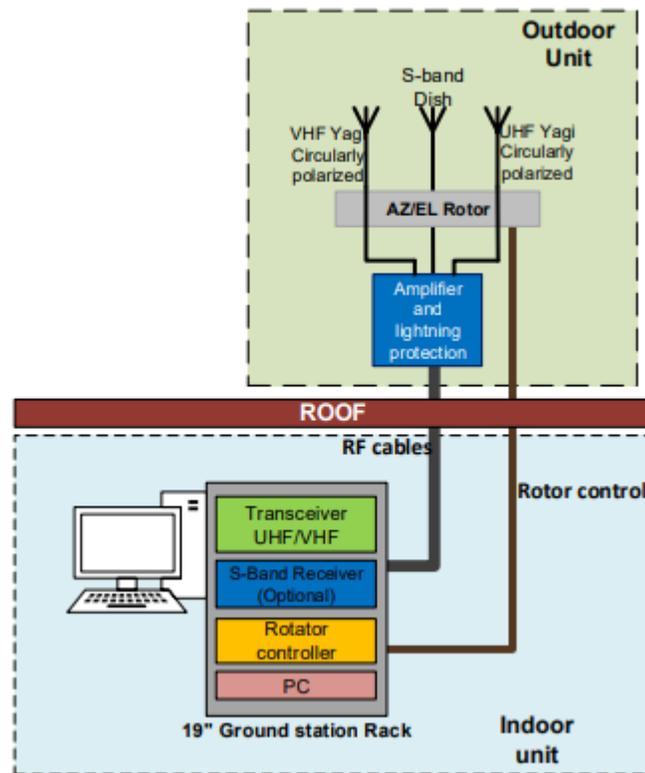


Figure 44: Indoor and outdoor unit diagrams [72]

In the set-up block diagram, there is both the outdoor and indoor components of the ground station. The outdoor unit is comprised of mainly the rotor and dish. The indoor unit contains the instruments such as the transceiver, controllers, and the PC. Figure 45 outlines the architecture of the software for this ground station [72].

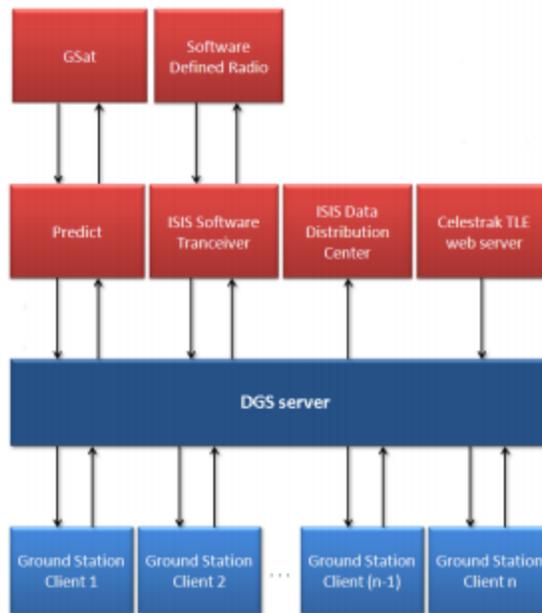


Figure 45: Software architecture [72]

The Distributed Ground Station (DGS) server is one of the main software components. The DGS and other software needed comes pre-installed on the PC in the kit with a GUI for configurations and control [72].

6.4 Access Time Calculation and Analysis

Based off the AWS Ground Station coverage, the team was able to determine the access times for the duration of MagSTARS' mission. Using the STK scenario, as described in Section 4.3, the selected ground station locations were input as "Place" objects. Each of these Place objects were assigned a child Sensor object. Each sensor was in turn assigned a Receiver object. These receivers were set to track the transceiver that was attached to the Satellite object from the original scenario. The transceiver on the satellite was given the parameters based on the transceiver component the team had selected for MagSTARS as described in Section 6.2. From here, the team

was able to run an access time analysis for the mission duration. Table 18 below shows the average calculated access time for each AWS ground station location.

Table 18: Access Times by Location

Location	Average Access Time (s)	Average Access Time (min)
Bahrain, Pakistan	433.422	7.22
Cape Town, South Africa	430.690	7.18
Columbus, Ohio, US	427.595	7.13
London, England	425.473	7.09
Salem, Oregon, US	429.768	7.16
Sau Paulo, Brazil	421.408	7.02
Sydney, Australia	425.960	7.10
Tokyo, Japan	421.696	7.03
Total Averages	427.0015	7.12

The times in the table are averages of each pass by MagSTARS calculated by STK. For a given ground station location, the access time could vary significantly. While the higher the access time the better, the lower access times could be problematic. In cases where the access time is significantly lower than the average, the data can be stored on a SD card within the iOBC. The data can be held here until the next suitable access time is reached. Table 19 below gives the total access time over the mission duration and the percent of coverage, on average for the mission. A sample of the full access report can be found in Appendix E.

Table 19: Access Time for Mission Duration

Total Access Time	126.33 hours	5.26 days
Total Mission Time	4584 hours	191 days
Percent of Mission Covered	2.7 %	

6.5 Results

From the STK scenario as described in Section 4 the team was able to analyze the access times for each ground station location along with general information about access periods over the scenario. Analysis of the effects of weather and interference on access time was performed. After running the simulation once more with the weather effects, it was noted that the effects do not change the access periods in a significantly enough way.

Further analysis was done in order to properly determine the effectiveness of the access times. An analysis and calculation of the data rates was performed. For transmission, the data rate is about 20 Mbps while for receiving the data rate is about 0.064 Mbps. An estimation of a typical file size that would be received by MagSTARS was about 0.01 Mb. For the data that would be transmitted, the files size was estimated based on the data rate of the payload which was about 0.0137 Mbps. The following table shows the file size estimation for data collected by the payload over a certain period of time.

Table 20: Data Rate Calculation

Rate (Mbps)	Time Collecting(s)	Size (mb)
0.0137	10	0.137
0.0137	20	0.274
0.0137	40	0.548
0.0137	80	1.096
0.0137	160	2.192
0.0137	320	4.384
0.0137	640	8.768
0.0137	1280	17.536
0.0137	5400	73.98
0.0137	10800	147.96
0.0137	37800	517.86

The bolded values in Table 20. indicate an estimate of how much data would be collected over the period on one orbit. Table 21 was then used to calculate the Table 22 that details the time needed to transmit the data collected by MagSTARS.

Table 21: Transmission Statistics

Transmitting Stats	
File Size (Mb)	TX Time (s)
0.137	0.00685
0.274	0.0137
0.548	0.0274
1.096	0.0548
2.192	0.1096
4.384	0.2192
8.768	0.4384
17.536	0.8768

The following table presents similar data for the data MagSTARS could receive. It starts with the initial estimation of the telemetry file size and then accounts for larger file sizes.

Table 22: Receiving Statistics

Receiving Stats	
File Size (Mb)	RX Time (s)
0.01	0.15625
0.1	1.5625
0.2	3.125
0.3	4.6875
0.4	6.25
0.5	7.8125
0.6	9.375
1	15.625

With the lowest access time of about 8.8 seconds, a limit can be determined for how much data can be transmitted or received safely during the access period. For transmitting data, the limit is around 147.96 Mb, equivalent to collecting data continuously for about 2 orbits. The receiving limit is 0.5 Mb. There is a small satellite delay time of 30-50 ms, which was deemed negligible as the average access times are generous enough for the data MagSTARS will handle [73].

From the team's analysis there is optimal time per each access to transmit and receive the estimated data the MagSTARS would be handling. The 8.8 second access time is a rarely occurring period and the average access time is about 7.12 minutes, which would allow well over the estimate that can be handled by MagSTARS. The average would allow about 8500 Mb of data to be transmitted or 27 Mb of data to be received. Based on the power analysis the team performed, it was estimated that the telecommunications equipment can run about 20% of the time during the

science phase. This phase will be where a majority of the data will be collected and transmitted. During the propulsion phases the telecommunications system will run less frequent and only run to receive data, every few orbits. As there is less data to manage during these times, this was deemed acceptable.

7 Attitude Determination and Control Subsystem

This chapter describes the theory, design, and simulation of MagSTARS' Attitude Determination and Control System. Actuator and sensor selection, control scheme development, and system simulation and testing are discussed.

7.1 Attitude Determination and Control System Overview

The purpose of the MagSTARS Attitude Determination and Control System (ADCS) is to ensure the CubeSat achieves the proper attitude for each leg of the mission. Without a well-developed ADCS, the power, communication, and science requirements of the mission could not be met. The primary requirement for the MagSTARS ADCS was to facilitate a constant rate spinning motion during scientific activities. To meet this requirement, the ADCS uses a suite of sensors and actuators to determine and correct the attitude of MagSTARS.

For attitude determination, MagSTARS combines data gathered from Sun sensors, a magnetometer, linear accelerometer and angular velocity sensor, and a Global Positioning System (GPS) receiver. Data from these sensors is used to produce an estimation of the CubeSat's position and attitude, as well as the magnetic field direction and strength. Using this state estimate, control commands are produced and actuated by three orthogonal magnetorquers. In the following sections, the theory, component selection methodology, design implementation, and control scheme design for this ADCS are described.

7.2 Attitude Determination and Control Component Selection

Selection of ADCS components was driven by the mission requirements and physical limitations of the CubeSat design. Three primary ADCS capabilities were identified as detumbling,

attitude determination and target pointing, and science mission tumbling motion. To facilitate each of these capabilities, sensors and actuator types were identified to form a full ADCS. The sensors selected were Sun sensors, a linear accelerometer, an angular velocity sensor, a GPS receiver, and a magnetometer. For actuation, a three axis magnetorquer set was chosen. Finally, it was determined that an onboard computer would be necessary for data management and actuation control. Following the identification of these components, multiple options for each were identified and compared.

Having identified the necessary components for the ADCS, multiple options of each were identified for comparison. The characteristics of each individual component were recorded with a focus on mission critical characteristics such as mass, power requirements, and temperature range. To facilitate selection of individual components for use on MagSTARS, the team developed decision matrices for each component type. For each component, the three characteristics listed above, as well as additional characteristics relevant to that component, were scored from one to five. A score of five was given to the component or components that performed the best with lower scores assigned to components that did not perform as well. Additionally, the estimated importance of each characteristic to a component's overall performance was weighted either from one to five or one to eight if greater resolution was needed. While both the component scoring and weighting were subjectively decided upon, they were based on an understanding of what performance levels were desirable for MagSTARS. The total score received by each component was determined through summing the multiples of each characteristic score and weight. Table 23 shows a sample decision matrix setup in the manner used by the team.

Table 23: Sample Decision Matrix

Component Type					
Factor	Mass	Power	Thermal	Resolution	Total
Weight	4	3	2	5	
Component A	3	5	2	3	46
Component B	3	2	5	3	43
Component C	5	4	3	5	63

Using this method, the sensors, actuators, and onboard computer for the ADCS that would best meet the mission requirements for MagSTARS were selected.

7.2.1 Magnetorquer Theory and Selection

A magnetorquer (also known as magnetic torquer or torque rod) is an electromagnetic device used to provide torques for the control of satellites [74]. Magnetorquers work by producing a magnetic dipole which interacts with the Earth’s magnetic field to produce a torque. To produce a magnetic dipole, current is passed through an electromagnetic coil. The magnetic dipole, denoted by \vec{m} , can be expressed as a multiple of the number of coil turns n , the current I , and the coil vector area \vec{A} :

$$\vec{m} = nI\vec{A} \quad (7)$$

The coil vector area direction is defined by the right-hand rule. The fingers curl in the direction of the current and the thumb points in the direction of the coil vector. When a magnetic

dipole interacts with a magnetic field, \vec{B} , a torque, $\vec{\tau}$, is created which is orthogonal to both vectors as defined by the following equation [75]:

$$\vec{\tau} = \vec{m} \times \vec{B} \quad (8)$$

The direction and magnitude of the torque generated by the magnetorquer depends on the direction and magnitude of the local magnetic field. Therefore, a measurement or model must be used to determine the local magnetic field. With three orthogonal magnetorquers and knowledge of the local magnetic field, full orientation control is possible. If one of the magnetorquer axes is aligned with the magnetic field, controlled rotation about that axis will not be possible until the magnetorquer axis is moved out of alignment with the magnetic field [75].

There are three main varieties of magnetorquers. Although they are characterized by different structural designs, the theory behind their operation is the same. The first variety is an embedded magnetorquer. Embedded magnetorquers use the copper laminate in a printed circuit board to form a loop. Due to this design, embedded magnetorquers generally are not able to produce magnetic dipoles of equal strength to those produced by the other two varieties. The second variety is known as a torque rod and consists of a long and narrow rod with a high count of wire turns to compensate for the low coil vector area. The third type, air core magnetorquers, use a wide but short structure to produce a large coil vector area. Two orthogonal torque rods are frequently paired with an air core magnetorquer to produce a full magnetorquer set as shown in Figure 46 [75].



Figure 46: NanoAvionics Magnetorquers MTQ3X full magnetorquer set employing two torque rods and one air core magnetorquer [72]

Magnetorquers are typically limited to use on smaller satellites like CubeSats due to the limited torque they can produce without requiring excessive amounts of power.

While researching potential magnetorquers for MagSTARS, the team decided that it was only interested in considering fully integrated magnetorquer systems. Such systems are equipped with control boards from the factory and have been tested extensively by their manufacturers. Creating a full three axis set from individual magnetorquers would significantly increase complexity and possibility of failure. Following this decision, the team identified two possible magnetorquer sets for use on MagSTARS. Using a decision matrix, shown in Appendix F, the team identified the NanoAvionics Magnetorquers MTQ3X as providing the best characteristics for meeting the mission requirements. The specifications of the MTQ3X are shown in Table 24.

Table 24: NanoAvionics Magnetorquers MTQ3X Specifications [73]

Parameter	MTQ3X
Mass	205 g
Power (Typical consumption)	400 mW
Thermal	-40 to +85°C
Dimensions	96 x 94 x 17 mm
Magnetic Dipole Strength	0.3 Am ² XY axes, 0.34 Am ² Z axis

7.2.2 Sensor and Onboard Computer Selection

As part of the attitude determination system, MagSTARS is equipped with six Sun sensors, one for each face of the CubeSat. Sun sensors can identify the position of the Sun when within the Sun sensor's field of view, typically ranging from 90 to 180 degrees. One sun sensor was used for each face of CubeSat to ensure that a sun direction measurement could be made from any attitude. The ADCS will use these Sun sensors during all phases of the mission with short interruptions during periods that MagSTARS is eclipsed by the Earth.

Following detumbling, MagSTARS will always be in an attitude with the solar panel equipped 6U face normal vector parallel or near parallel to the Sun vector. The solar panels selected for MagSTARS, discussed in Section 5, each had a built-in sun sensor. This meant that only two additional sun sensors were needed to cover all six faces. A decision matrix, shown in Appendix F, was employed to compare Sun sensor options and make a selection. The NewSpace Systems NCSS-SA05 was chosen for its higher performance and low mass, volume, and power requirements. The NCSS-SA05 is shown in Figure 47.



Figure 47: NewSpace Systems NCSS-SA05 [76]

As the specifications for the built-in solar panel sun sensors were not available, it was assumed that the performance of these sun sensors was equivalent to that of the NCSS-SA05. The specifications of the NCSS-SA05 are included in Table 25 below.

Table 25: NewSpace Systems NCSS-SA05 Specifications [76]

Parameter	NCSS-SA05
Mass	<5 g
Power (Typical Consumption)	<50 mW
Thermal	-25 to +70 °C
Field of View	114 degrees
Root Mean Square Error	<0.01 degrees

For estimating its position and velocity in its orbit, MagSTARS utilizes a GPS receiver. The GPS data is critical to the success of multiple mission aspects such as the TRIAD method for attitude determination, discussed more in Section 7.3.2, and the planning and execution of

maneuvering burns. Therefore, identification of a high-performance GPS receiver was a key factor in developing the ADCS.

Three GPS receivers were identified as options and compared through the decision matrix system with a focus on mass, thermal, power, size, and accuracy. From a comparison of these factors, the Hyperion Technologies GNSS200 receiver was selected for use on MagSTARS. The GNSS200 is shown below in Figure 48.

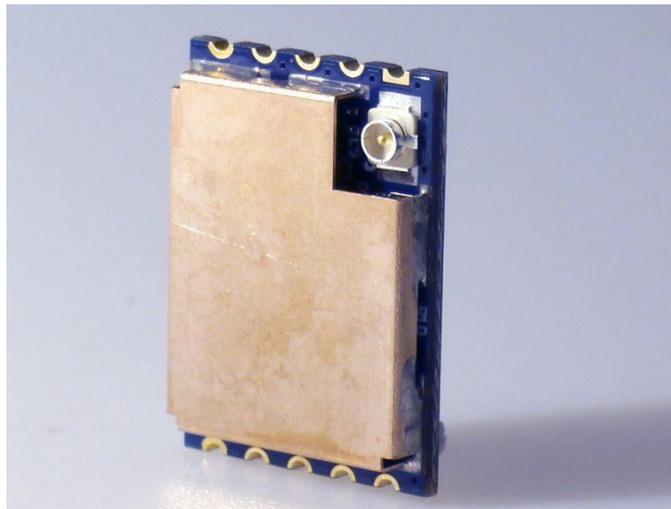


Figure 48: Hyperion Technologies GNSS200 [77]

One parameter of interest, velocity accuracy, was not available for the GNSS200. Therefore, it was assumed that it performed similarly to the other receivers in this area. If new data showed that the GNSS200 performed significantly worse in this area than expected, the decision matrix would have to be rescored and reevaluated. The specifications of the GNSS200 are shown in Table 26

Table 26: Hyperion Technologies GNSS200 [75]

Parameter	GNSS200
Mass	3 g
Power (Typical Consumption)	157 mW
Thermal	-40 to +85 °C
Size	20 x 14.5 x 3.1 mm
Position Accuracy	8 ³ m
Velocity Accuracy (Assumed)	25 ³ cm/s

The decision matrix used for the selection of the GNSS200 can be found in Appendix F.

An additional sensor set used on MagSTARS is a linear accelerometer and angular velocity sensor. In addition to the measurements they directly provide, the data from these sensors can be numerically integrated to produce improved estimates of the state of the CubeSat. For example, linear accelerometer data and GPS data are used to produce a more accurate velocity estimate, which allows for more accurate burns. Similarly, angular velocity data can be used to improve an estimate of the attitude. Angular velocity data is also central to the ADCS as it is used to determine attitude control torque demands, as discussed in Section 7.3.

Using a decision matrix, the BEI Model 4311A linear accelerometer and Applied Technologies ARS-15 angular velocity sensor were selected. These sensors met the constraints set by the mission requirements and outperformed the other sensors. The Model 4311A data used did not include dimensions. However, as it was the lowest mass linear accelerometer of the three BEI models analyzed, it was assumed the size would be equal to if not smaller than the second lowest mass model. The Model 4311A and ARS-15 are show in Figure 49 and Figure 50 respectively.

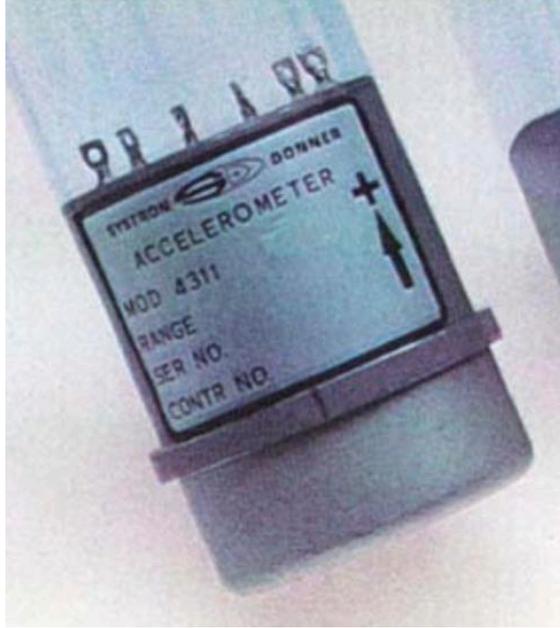


Figure 49: BEI Model 4311A linear accelerometer [78]



Figure 50: Applied Technologies ARS-15 [77]

The Model 4310 and ARS-15 both provide high accuracy, high precision measurements with lower mass and power requirements. The specifications of each are listed in Table 27.

Table 27: Applied Technologies ARS-15 and BEI Model 4311A [78], [79]

Parameter	ARS-15	Model 4311A
Mass	60 g	34g
Power (Typical Consumption)	<300 mW	150 mW
Thermal	Unknown	-40 to +93 °C
Size	33.81 x 19.81 x 19.81 mm	Unknown
Range	±10 rad/s	±0.5 gees
Resolution	<2.5×10 ⁻⁵ rad/s	<2.5×10 ⁻⁴ gees

The decision matrices used to select the Model 4311A and ARS-15 can be found in Appendix F.

With its wide array of sensors and components, MagSTARS requires a central onboard computer for data processing, power distribution, and attitude control calculations. Given these requirements, an onboard computer with multiple data inputs and outputs, high processing speed, and built-in data processing capability was desired. Using these desired traits, viable options were identified and compared with a decision matrix. Through this method, the Innovative Solutions In Space (ISIS) On-Board Computer. (iOBC) was selected. The iOBC is shown below in Figure 51



Figure 51: ISIS On-Board Computer without daughterboard [78]

The specifications of the iOBC are shown in Table 28 below.

Table 28: ISIS On-Board Computer [80]

Parameter	iOBC
Mass	100 g (with daughterboard)
Power (Typical Consumption)	400 mW
Thermal	-25 to +65 °C
Size	96 x 90 x 12.4 mm
Processing Speed	400 MHz
Storage	Up to 2x32 GB SD cards
Analog to Digital Converter	10 bit, 8 channels

The decision matrix used to select the ISIS On-Board Computer can be found in Appendix

F.

To produce proper control responses for attitude control and to enable attitude determination, MagSTARS must be equipped with a magnetometer. A magnetometer is a device used to detect the magnitude and direction of a local magnetic field. To meet the requirements of MagSTARS, several magnetometers were compared using a decision matrix. From the magnetometers analyzed, the Spacemanic LODESTONE SM-ADS-AG4 scored highest in the decision matrix. This was due to its high performance despite a low mass and power consumption. Additionally, the small size of the SM-ADS-AG4 was beneficial as it enabled mounting far from the magnetorquers. This would reduce the amount of magnetic field noise induced by actuating the magnetorquers. The SM-ADS-AG4 is shown below in Figure 52.

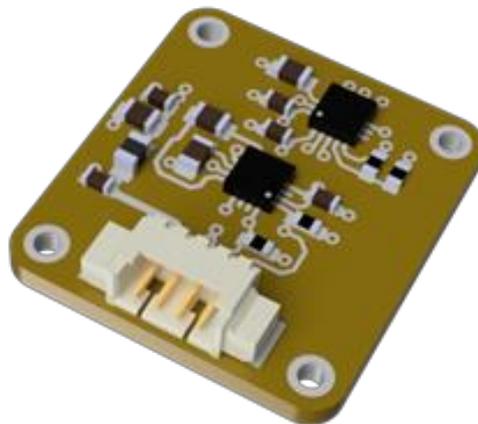


Figure 52: Spacemanic LODESTONE SM-ADS-AG4 [79]

The specifications of the magnetometer are shown in Table 29.

Table 29: Spacemanic LODESTONE SM-ADS-AG4 [81]

Parameter	SM-ADS-AG4
Mass	2 g
Power (Typical Consumption)	10.4 mW
Thermal	-40 to +85 °C
Range	±8 G
RMS Noise	0.4 mG

The decision matrix used to select the SM-ADS-AG4 is shown in Appendix F.

Having completed selection of the components used for the ADCS, the team was able to move into performing simulations of component behavior, attitude determination, and attitude control.

7.3 Attitude Determination and Control Methods and Simulations

The following sections discuss MagSTARS' attitude determination and control schemes developed to employ the components selected above. Control schemes for detumble, targeted pointing, and science mission tumble are discussed in addition to attitude determination methods. For each, the physical and mathematical theory, the computational implementation, and simulation are described.

7.3.1 Detumble Methods and Simulations

Following deployment from the CSD, it is likely that MagSTARS will have some initial rotation induced by the ejection. To successfully complete its mission, MagSTARS must be able to remove this initial angular velocity in an action known as detumbling. Detumble control

methods are dependent on satellite design and control actuator selection. For magnetorquer equipped satellites, a common controller of choice is the B-dot controller. This controller drives the angular rates of a satellite to zero but is not able to achieve pointing goals. Therefore, the standard B-dot controller could not be used for all phases of the mission as it is unable to produce the desired tumbling motion for MagSTARS' science phase.

The B-dot controller was first proposed in 1972 by Seymour Kant and operates by generating torque commands based on the magnetic field state [82]. In the general form of the B-dot controller, the time derivative of the magnetic field is used to define the output magnetorquer dipole moment direction. The general form, where \bar{b} is the unitized magnetic field vector and k the controller gain, is written as follows:

$$\bar{m} = -\frac{k}{\|\bar{b}\|} \dot{\bar{b}} \quad (9)$$

As the time derivative of the magnetic field is not directly measured by a magnetometer, modifications to the control law can be applied to enable the use of direct spacecraft measurements. A standard assumption used in B-dot controller design is that the time derivative of the magnetic field in the Earth fixed frame is much smaller than the angular velocity between the body and Earth frames. Following this assumption, the controller can be rewritten to use the body frame angular velocity, which is directly measured. The full derivation of this modification is discussed by the 2017 MQP team of Agolli, Gadoury, and Rathbun. The rewritten control law is expressed in the following form [21]:

$$\bar{m} = \frac{k}{\|\bar{b}\|} \bar{\omega} \times \bar{b} \quad (10)$$

This form of the control law uses angular velocity, ω , and magnetic field vector information, both of which are directly measured by components on MagSTARS. To simulate

detumbling using the B-dot controller, a MATLAB script, shown in Appendix G, was produced for use with the STK Attitude Controller. The Attitude Controller allows selection of initial pointing and angular rates for simulations. The Attitude Controller sends desired information, in this case the angular rates and magnetic field vector, through the MATLAB script which outputs control torques through use of the B-dot controller. STK applies the calculated output torques to the satellite and simulates its motion using a numerical integration method. To ensure the simulated satellite motion was accurate, the moment of inertia tensor acquired through CAD modeling of MagSTARS was input into the STK scenario.

The effectiveness of the B-dot controller was dependent on expected initial angular rates. With higher angular rates would come increased detumble times as well as increased energy usage. Therefore, determining representative initial angular rates was identified as a key factor in producing valid detumble simulations. Simulation of CSD ejection events was not desirable as this would be highly complex and time consuming. Instead, the team decided to select initial conditions based on tests conducted by the dispenser manufacturer, Planetary Systems Corporation (PSC).

The PSC team conducted tests of 6U payloads being ejected from a CSD on parabolic aircraft flights to simulate the microgravity environment of low Earth orbit. The CSD used by the PSC team was the same model selected for use with MagSTARS. They measured pre- and post-ejection angular rates of the payloads to determine the rates induced by the CSD. Over 100 tests were performed producing a large data set of initial rotation rates. Data collected from these tests showed that initial tumble rates typically did not exceed ± 10 deg/s about the payload axes [31]. Based off this research, initial tumble rates within these values were selected for testing. Figure 53 shows a plot of the initial tumble rates measure by the PSC team.

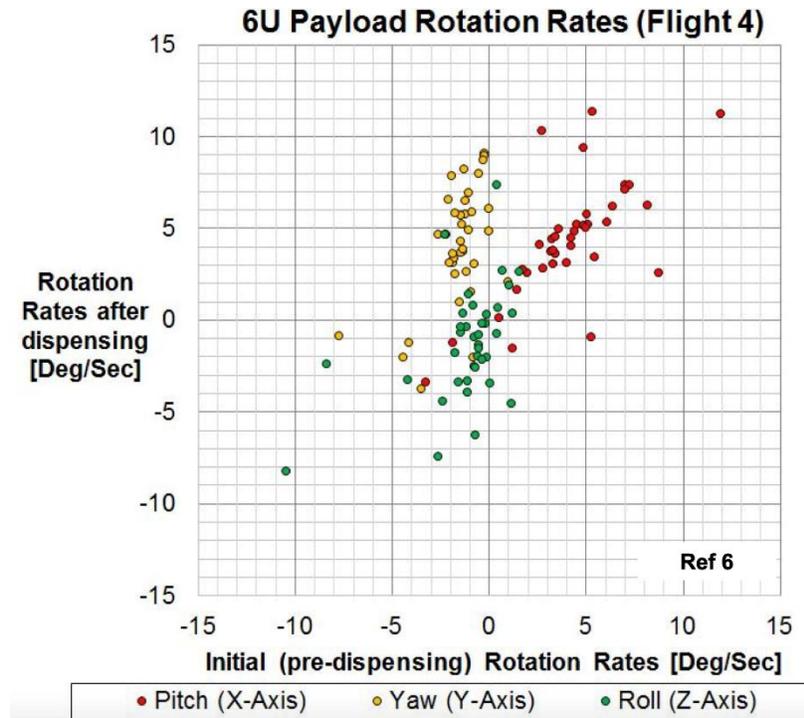


Figure 53: 6U payload initial tumble rates about each body axis [31]

Next, the B-dot controller was implemented into a MATLAB script for use with the STK Attitude Simulator tool. To ensure the B-dot controller was working as expected, multiple tests were performed with various initial conditions of angular rates and location in orbit. The detumble time varied depending on the initial conditions used but did not exceed 50 minutes for any cases simulated. In Figure 54 three detumble cases for large initial angular rates are shown.

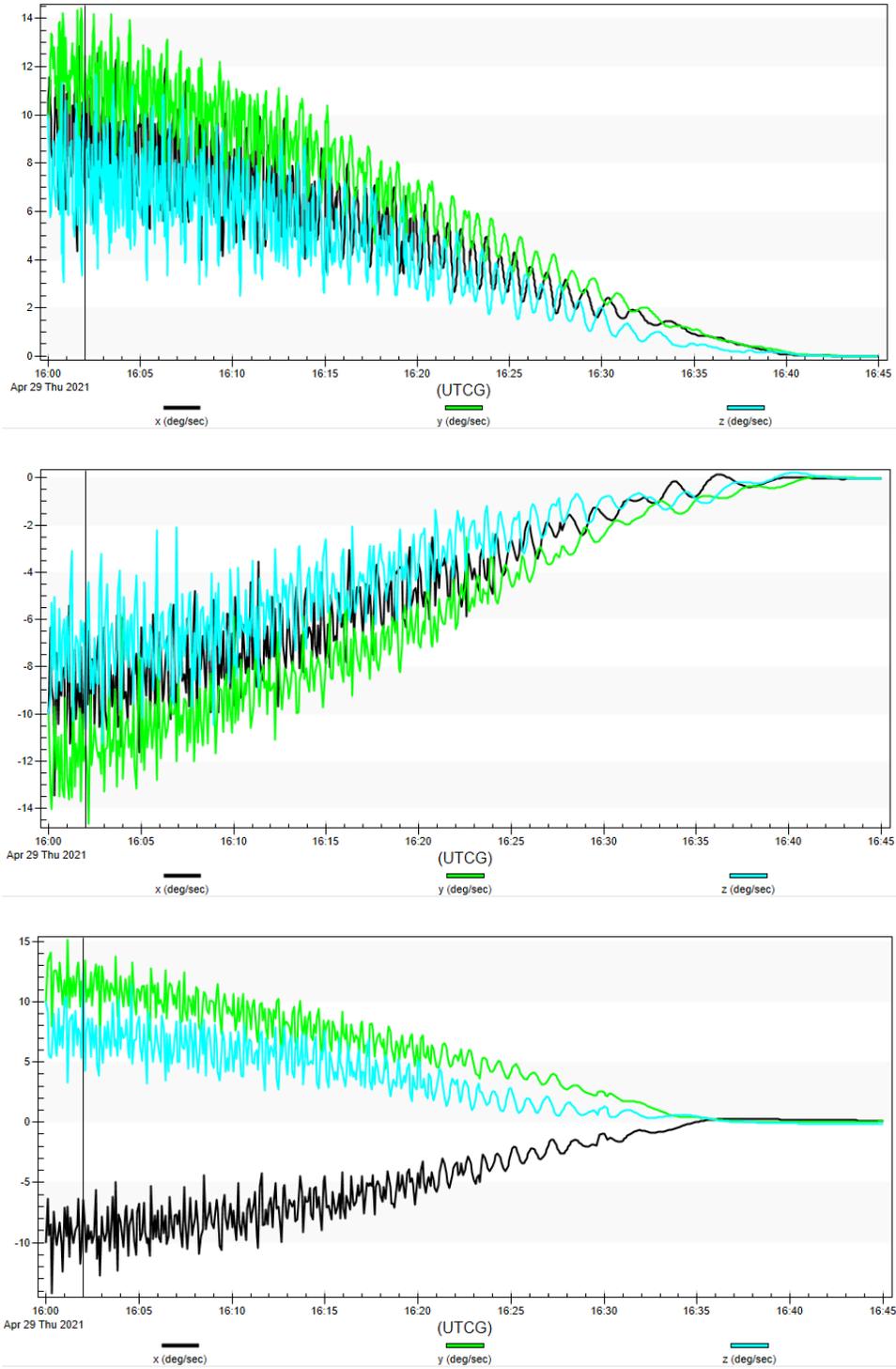


Figure 54: Three detumble simulations from large angular rates

Following the detumble simulations, the B-dot controller development was completed, and the MATLAB script was finalized. The B-dot controller method developed for MagSTARS was successful in reducing initial angular rates induced by deployment to zero. After detumbling, MagSTARS needed to determine its attitude, as described in the following section.

7.3.2 TRIAD Method and Simulation

The TRIAD method is one of the earliest and simplest methods for the determination of a spacecraft's attitude. It was developed by Harold Black who first published the method in a paper in 1964 [83]. The TRIAD method employs two [84] reference vectors which are used to produce a direction cosine matrix (DCM) that provides an estimate of the spacecraft's attitude [83]. The two reference vectors used by MagSTARS are the Sun vector and the magnetic field vector. These vectors are obtained from the onboard Sun sensors and magnetometer and are therefore subject to sensor noise. Despite this induced error, the TRIAD method can produce a DCM using the following mathematical manipulation.

Begin by defining two vectors, \overline{R}_1 and \overline{R}_2 , as the true Sun and magnetic field vectors in the Earth Fixed Inertial (ECI) frame. Next, assume the ECI and body frames are located at the same origin to simplify calculations. Two observation vectors, \overline{r}_1 and \overline{r}_2 , define the noisy Sun and magnetic field vectors measured in the body frame. From these vectors, unit vectors \hat{S} , \hat{M} , \hat{s} , and \hat{m} are defined:

$$\hat{S} = \frac{\overline{R}_1}{\|\overline{R}_1\|} \quad (11)$$

$$\hat{S} = \frac{\bar{r}_1}{\|\bar{r}_1\|} \quad (12)$$

$$\hat{M} = \frac{\bar{R}_1 \times \bar{R}_2}{\|\bar{R}_1 \times \bar{R}_2\|} \quad (13)$$

$$\hat{m} = \frac{\bar{r}_1 \times \bar{r}_2}{\|\bar{r}_1 \times \bar{r}_2\|} \quad (14)$$

Using these vectors, a DCM \mathbf{A} that converts body fixed vectors to ECI vectors is defined with the following matrix multiplication:

$$\mathbf{A} = [\hat{S} \ \hat{M} \ \hat{S} \times \hat{M}] [\hat{s} \ \hat{m} \ \hat{s} \times \hat{m}]^T \quad (15)$$

Lastly, the DCM may be converted to a quaternion or Euler angle vector as an estimate of the satellite's attitude. Due to the radius of MagSTARS' orbit having been much smaller than the Earth to Sun distance, it is assumed that the direction of the Sun vector is independent of the position of MagSTARS in its orbit. This assumption may produce a maximum error of approximately 2.5×10^{-3} degrees. This is acceptably low as it is an order of magnitude less than the Sun sensor error. The magnetic field vector however is entirely dependent on the location of MagSTARS. Therefore, to produce an estimate of the magnetic field vector in the ECI frame, GPS positional measurements must be used. These measurements are also subject to noise which produces additional error in the attitude estimate.

With the method developed, TRIAD was implemented in a MATLAB program, shown in Appendix H, and initial testing was done to ensure the program was producing accurate results.

An example implementation of the TRIAD method with and without artificial sensor noise is shown below:

$$\begin{aligned}
 \overline{R_1} &= \begin{bmatrix} 1 \\ 0 \\ 0 \end{bmatrix} \\
 \overline{r_1} &= \begin{bmatrix} 0 \\ \frac{\sqrt{2}}{2} \\ \frac{\sqrt{2}}{2} \end{bmatrix} \\
 \overline{r_{1,error}} &= \begin{bmatrix} 0 \\ \frac{\sqrt{2}}{2} + 0.01 \\ \frac{\sqrt{2}}{2} + 0.02 \end{bmatrix} \\
 \overline{R_2} &= \begin{bmatrix} 12 \\ 0 \\ 25 \end{bmatrix} \cdot 10^{-6} \text{ mT} \\
 \overline{r_2} &= \begin{bmatrix} 25 \\ 12 \frac{\sqrt{2}}{2} \\ 12 \frac{\sqrt{2}}{2} \end{bmatrix} \cdot 10^{-6} \text{ mT} \\
 \overline{r_{2,error}} &= \begin{bmatrix} 24.98 \\ 12.01 \frac{\sqrt{2}}{2} \\ 11.96 \frac{\sqrt{2}}{2} \end{bmatrix} \cdot 10^{-6} \text{ mT} \\
 &= \begin{bmatrix} 0 & 0.707 & 0.707 \\ 0 & -0.707 & 0.707 \\ 1 & 0 & 0 \end{bmatrix} \\
 A_{error} &= \begin{bmatrix} 0 & 0.702 & 0.712 \\ 0.004 & -0.712 & 0.702 \\ 0.999 & 0.003 & -0.003 \end{bmatrix}
 \end{aligned} \tag{16}$$

This example shows the TRIAD method can produce highly accurate attitude estimates despite the inclusion of sensor error. Having proven that the TRIAD method MATLAB program was working properly, it was then implemented in the attitude control schemes discussed in the following section.

7.3.3 Proportional-Derivative and Trajectory Optimization Controllers

Though the B-dot controller discussed earlier could reduce the angular rates of MagSTARS to zero, it was not capable of producing desired attitudes or non-zero angular rates. These limitations meant that a different controller was necessary for achieving the targeted pointing and science mission tumbling motion, both of which were critical for the successful operation of MagSTARS. Initial development focused on developing a proportional-derivative (PD) controller for this purpose. However, it was eventually determined that such a controller would not be able to achieve desired attitude results due to the behavior of the magnetorquers. An alternative method, using trajectory optimization from a 2019 paper by Gatherer and Manchester, was identified and developed [84]. Both methods will be described below.

A PD controller attitude control method was initially selected because of the simplicity of the controller development. A PD controller is a simplified version of a full proportional-integral-derivative controller. The integral term was deemed unnecessary as no external torques were modelled to act on MagSTARS. The PD controller uses measurements of attitude and angular rates as well as knowledge of the desired state to produce desired output torques. To ensure stability of the satellite motion, the proportional and derivative terms have gains that adjust their magnitude.

For initial testing, a basic PD controller was developed, as shown in Appendix I. This controller would produce an output torque of the same magnitude of the magnetorquers (10^{-5} Nm).

Additionally, the direction of the output torque could be in any direction. This was not realistic as the torque produced by the magnetorquers must be orthogonal to the Earth's local magnetic field vector. The form of this basic PD controller was as follows:

$$\tau_{1:2:3} = -I_{1:2:3}T_1 - I_{4:5:6}T_2 - I_{7:8:9}T_3 \quad (17)$$

Where I is the moment of inertia matrix and T is defined as:

$$T_{1:2:3} = kq_{d,1:2:3}q_{d,4} - c\omega_{1:2:3} \quad (18)$$

Where k is the proportional gain, c the derivative gain, q_d the desired attitude quaternion, and ω the angular velocity. Gain values of $k=0.001$ and $c=0.05$ were selected through repeated testing. Lastly, a clamping term was added to ensure that the output torque never exceeded 1.5×10^{-5} Nm. This term ensured that the basic PD controller torques would be of the same magnitude as those produced by the true magnetorquers. Figure 55 shows an example of this basic PD controller successfully changing the attitude of MagSTARS to achieve a desired state.

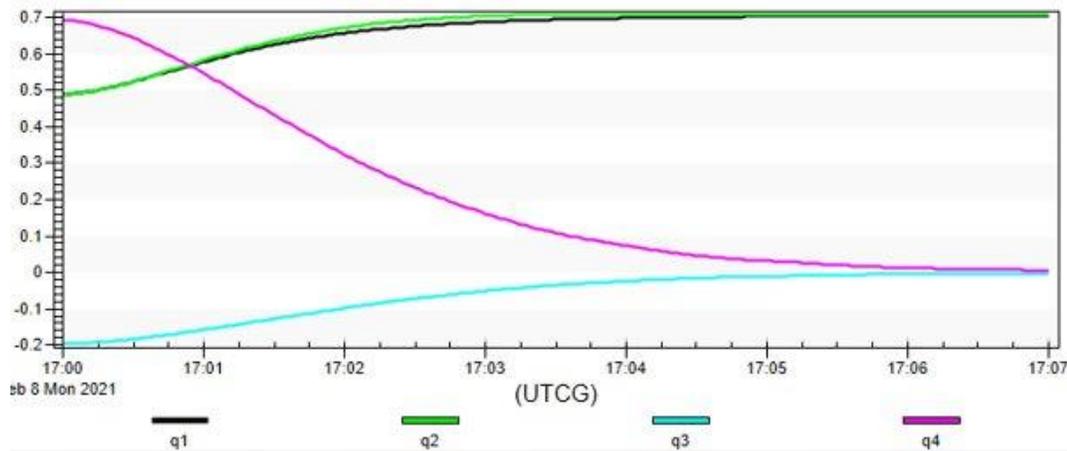


Figure 55: Basic PD controller attitude adjustment

As stated previously, this PD controller was not valid as the output torque was frequently not orthogonal to the Earth's magnetic field vector. In order to produce valid output torques, two modifications to the PD controller were made. First, the desired output torque vector produced

by the basic PD controller was projected onto the plane defined by the B vector. This ensured that the direction of the output torque was valid. Secondly, a magnetorquer dipole moment vector that could produce this output torque while meeting the performance limitations of the magnetorquer set was found. This was done with knowledge of the desired orthogonal output torque, $\bar{\tau}_o$, the Earth's magnetic field, and (9):

$$\bar{m} = \frac{1}{\|\bar{B}\|^2} \bar{B} \times \bar{\tau}_o + \lambda \bar{B} \quad (19)$$

Where lambda can be any positive or negative factor. This indicates that there are an infinite number of solutions for the dipole moment vector that would produce the desired output torque. Most solutions would not meet the physical limitations of the magnetorquers, so a range of lambda values were tested to find a valid solution. An example is shown below:

$$\begin{aligned} \bar{B} &= \begin{bmatrix} 22 \\ 14 \\ -9 \end{bmatrix} \cdot 10^{-6} \text{ Tesla} \\ \bar{\tau}_o &= \begin{bmatrix} -7.28 \\ 7.55 \\ -6.07 \end{bmatrix} \cdot 10^{-6} \text{ Nm} \\ \bar{m}_a &= \begin{bmatrix} -0.02 \\ 0.26 \\ 0.35 \end{bmatrix} \text{ Am}^2, \lambda = 0 \\ \bar{m}_b &= \begin{bmatrix} 0.02 \\ 0.29 \\ 0.33 \end{bmatrix} \text{ Am}^2, \lambda = 2 \cdot 10^3 \end{aligned} \quad (20)$$

Both solutions produce the desired output torque however only the \bar{m}_b vector can be produced by the magnetorquers. It should be noted that a large λ value was used because the magnitude of \bar{B} is much smaller than the magnitude of m .

With this modification put in place, the PD controller respected the physical limitations of the magnetorquers. The modified PD controller form is shown in Appendix J. However, testing showed that these modifications led to a failure of the controller to achieve desired attitudes. In Figure 56 the PD controller behavior for the same initial conditions used in Figure 55 is shown.

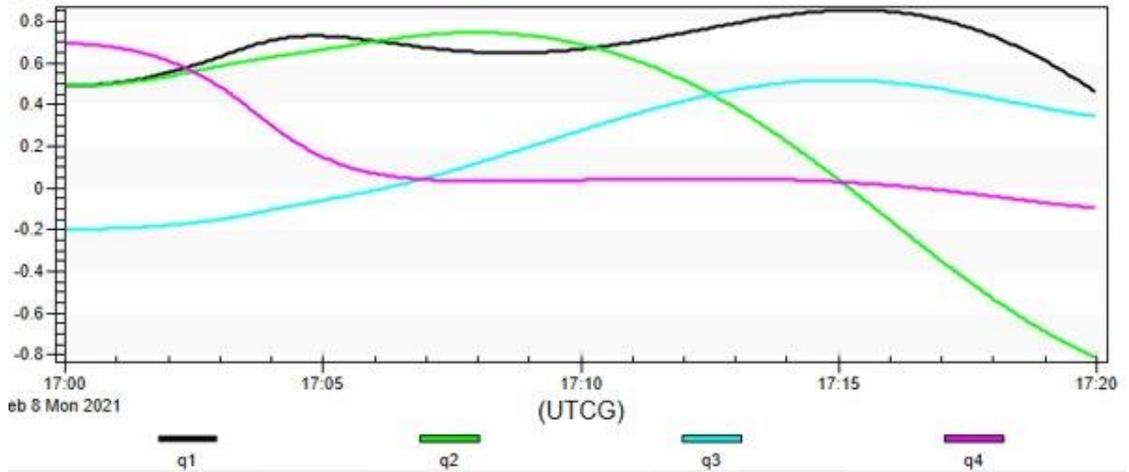


Figure 56: PD controller failure to achieve desired attitude

Given the complexities with torque limitations of magnetorquers and the inability of the PD controller to successfully deal with these limitations, an alternative control scheme was investigated. The magnetorquer only control problem has seen multiple approaches over multiple years. Improved mathematical and computational methods have enabled significant improvement in solving the nonlinear control problem. A 1996 paper looking at magnetorquer control schemes described the nonlinearity as being “a serious obstacle for using magnetorquer based control for three-axis attitude control” [85]. A new paper from 2019 presents a novel approach to this problem using trajectory optimization [84]. Results from this paper may be used for further control system development for MagSTARS.

8 Space Environment and Effects

This section describes the ambient space environment MagSTARS will be exposed to during its mission and its possible effects, including atmospheric drag from neutral and plasma species, spacecraft charging due to the plasma environment, possible impacts on sensors from the ambient magnetic field, and radiation impacts from background radiation.

8.1 The Ionospheric Environment

MagSTARS will spend the duration of its mission in the ionosphere, an upper layer of Earth's atmosphere characterized by shifting levels of ionized particles that have a variety of effects on spacecraft.

8.1.1 Ionosphere Overview

Earth's ionosphere is a layer of the atmosphere that ranges from approximately 50 km to 500 km in altitude. Ionization, recombination, diffusion, and gravity drive the ionosphere's particle composition. X-rays and ultraviolet light from the Sun cause electrons to break off normally neutral gas particles, ionizing these gasses. This process creates a region of ions and free electrons, or plasma. These ions and electrons can collide and recombine to form neutral gas particles, but in the daylight when the energy from the Sun is highest, the ionization process occurs faster than the recombination process, forming layers of high ion concentration. At night, in the absence of the Sun's energy, recombination takes over and ionization levels decrease. The particles of the ionosphere are bound to the atmosphere by Earth's gravitational force, which causes higher particle density at low altitudes. Combined with the ionization activity caused by the Sun, gravity

creates pressure gradients within the ionosphere. The particles diffuse from areas of high density to low density, causing constant fluctuations in the ionosphere's density and makeup [86].

The ionosphere is an extremely important layer of the atmosphere because of its impact on radio communication and GPS. The ions and electrons in the ionosphere reflect radio waves, allowing signals to be transmitted without a line-of sight. This concept is illustrated in Figure 57. Much modern technology, especially within the aerospace industry, relies on radio and GPS communication, so it is very important to understand the composition and phenomena of the ionosphere [87].

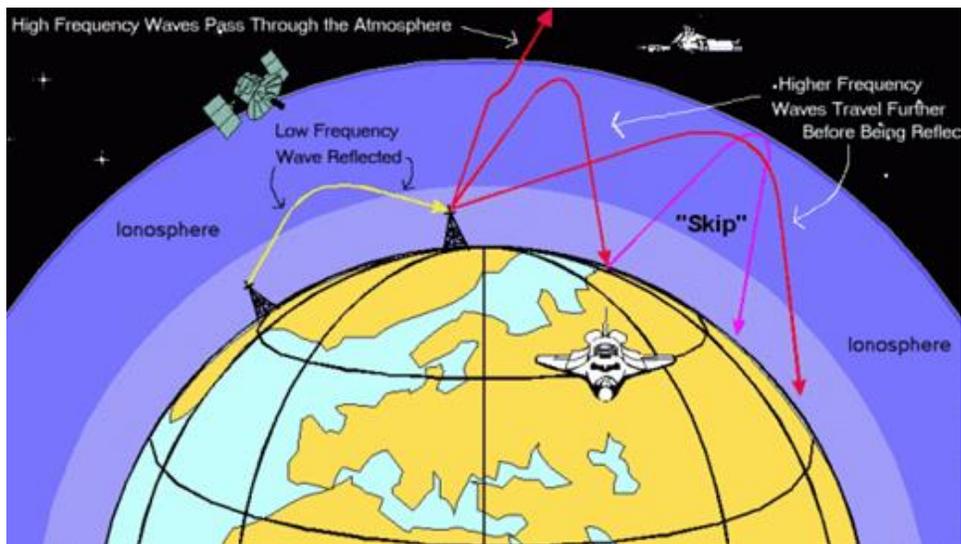


Figure 57: Radio communication's reliance on the ionosphere [87]

The ionosphere can be split into several "layers," which vary with season, time of day, and solar activity. Figure 58 shows these layers, as a function of electron density and altitude, during daytime and nighttime. Since the ionization process is driven largely by photoionization from solar radiation, electron densities decrease at night when solar intensity is lowest. The minima (night) and maxima (day) electron densities are represented by the shaded regions bordered by the green and red lines. The width of the solar minimum and maximum regions is due to seasonal and yearly

variations in solar intensity and conditions. For altitudes below approximately 300 km there is overlap on the electron density, which shows the wide fluctuation of electron density and dependence on changing environmental conditions.

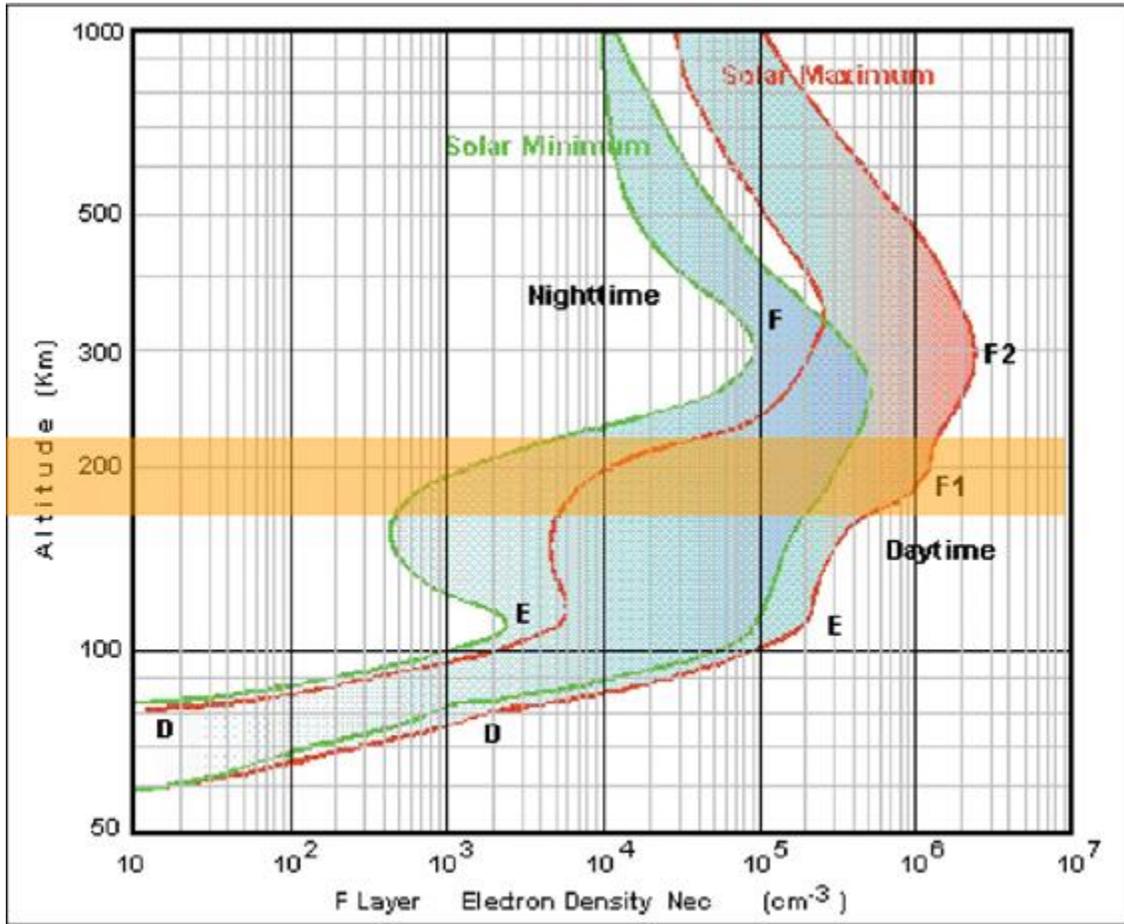
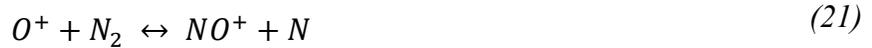


Figure 58: Electron density of the ionosphere, highlighted F1 region [88]

The D layer is the lowest altitude layer, with a relatively high density that enhances the recombination process, especially at night when temperatures are lower and solar radiation is not present. It ranges from approximately 50 to 90 km altitude during the day and essentially disappears at night when there is less direct radiation from the Sun. The E layer ranges from approximately 90 to 140 km and, like all layers, has lower ionization levels at night, though the

atmosphere is less dense than lower layers, allowing ionization to continue without direct radiation [89].

MagSTARS will orbit through the highest layer, the F layer, which begins at an altitude of approximately 140 km. During the day, the F layer splits into the F1 and F2 layers, with the F1 layer ranging from an altitude of approximately 140 to 200 km and the F2 layer approximately 200 km and above. This separation however, at approximately 200 km, varies with season and solar activity. The F layer is mostly comprised of oxygen (O, O₂, O⁺), Nitrosonium (NO⁺) ions, Nitrogen (N, N₂) and free electrons, with their concentrations governed by solar intensity and two main chemical processes:



The first process (Eq. (21)) converts atomic oxygen ions and nitrogen into molecular ions (NO⁺) and atomic nitrogen through a dissociative attachment reaction. The second process (Eq. (22)), converts the molecular ions to atomic nitrogen and oxygen through a recombination reaction with free electrons. These processes are governed by several factors, but mainly by the populations of oxygen ions and free electrons. Thus, at higher altitudes where the production of oxygen ions is high and free electrons abundant, the atomic oxygen governs the recombination process since free electrons quickly recombine with NO⁺. At lower altitudes, where the ionization rate is lower and electron population smaller, the recombination process is governed by the dissociative recombination (22)). At night, the quick recombination of electrons with the NO⁺ ions in the F1 layer cause it to effectively disappear as atomic oxygen product is significantly reduced. The F2 layer, however, is slower to fade because of the low amount of N₂ present which slows conversion

from O^+ to NO^+ and therefore slows the NO^+ /electron recombination process. This causes the F2 layer to remain present throughout the night, becoming the F layer [88].

8.1.2 Ionospheric Anomalies and Their Effects

Seasonal variation in electron density is also present in the ionosphere, as the Earth's distance from the Sun and the tilt of Earth's axis cause changes in the amount of solar energy that reaches the ionosphere. Seasonal variation is represented in Figure 58 by the span of densities covered in the solar minimum and solar maximum areas. Another known variation within the ionosphere is the Equatorial Ionization Anomaly, shown in Figure 59. This anomaly occurs in the F layer and is caused by Earth's electromagnetic field. The plasma in the lower F layer, which usually drifts horizontally, is driven upwards by Earth's electromagnetic field along field lines to higher altitudes, creating a "fountain" effect. This creates ionization crests, or areas of high ionization, displaced from the geomagnetic equator ($\pm 15^\circ$ latitude) and an area of low ionization at the equator [90]. However, this anomaly region is far from stable, and irregular plasma flow generates issues throughout the ionosphere. Aside from diurnal and seasonal variations, gravity and the electromagnetic field work against each other on the F layer plasma, creating perturbations in the plasma flow that can be very difficult to predict [88]. These perturbations create ionospheric scintillations, which are the rapid modification of radio waves caused by unexpected small scale "structures" in the ionosphere. Scintillations negatively impact radio communication and GPS signals and can cause errors on in data or complete loss of signal [91].

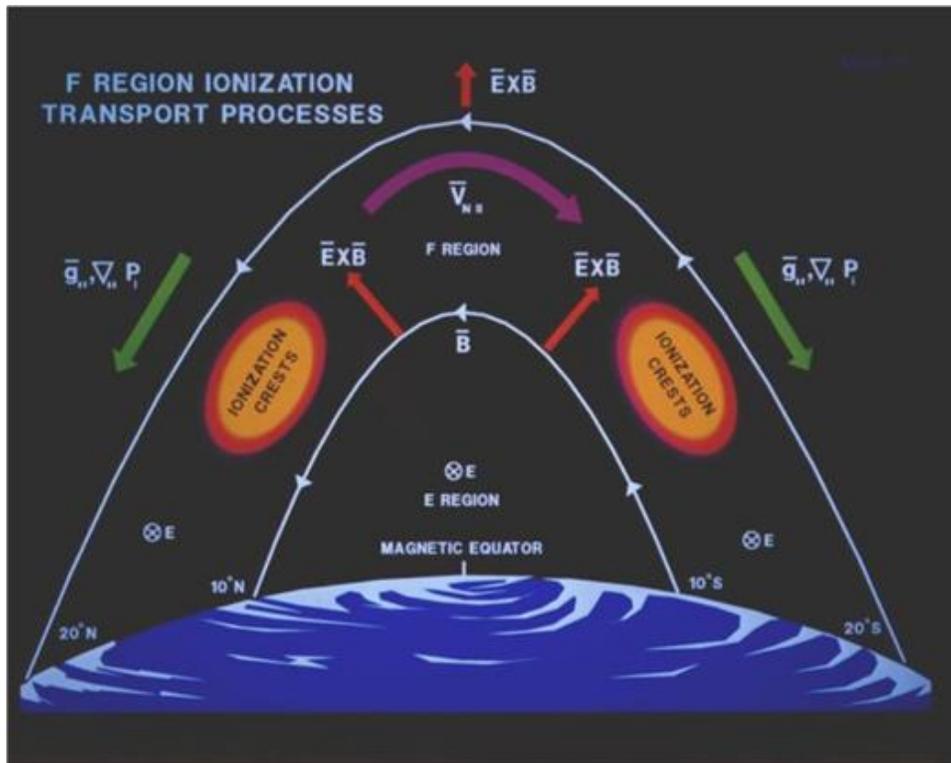


Figure 59: The Equatorial Ionization Anomaly [84]

Scintillation-causing perturbations in the ionosphere can be caused by a number of factors besides the Equatorial Ionization Anomaly, including Earth weather and solar activity [92]. MagSTARS' orbit will take it through a large variation of latitudes, including the Equatorial Ionization Anomaly. This will allow it to investigate the ionosphere's composition in a variety of environments and collect data that will give more insight into the variations of the ionosphere. These variations are widespread and difficult to model. Because so many crucial operations rely on understanding the ionosphere, it is important to have an accurate depiction of its composition. It is a goal of MagSTARS' mission to supplement existing ionospheric models with real, in-situ data in the hopes of furthering understanding of these complex variations.

8.1.3 Ionospheric Effects and Analysis

The neutral and plasma species that make up the ionosphere have several effects on the spacecraft that encounter them. These effects, including spacecraft charging and atmospheric drag, must be modeled and analyzed to ensure the mission will not be negatively impacted.

8.1.3.1 Atmospheric Drag

The neutral particles that fill the ionosphere coming into contact with MagSTARS as it moves through its orbit generate forces. The force along the direction of the spacecraft motion is referred to as atmospheric drag. The expected low altitude of MagSTARS' orbit means the atmospheric density is high, which increases the prospects of the drag force [40].

During the science leg of the mission, MagSTARS will encounter atmospheric densities ranging from $5.97 \times 10^{-11} \text{kg/m}^3$ to $2.62 \times 10^{-12} \text{kg/m}^3$ [93]. (23) was used to calculate the average drag force experienced by MagSTARS during the mission [94].

$$D = C_d \frac{\rho V^2}{2} * A \quad (23)$$

D is the calculated drag, C_d is the drag coefficient, chosen as 2.2, ρ is the average atmospheric density, $3.11 \times 10^{-11} \text{kg/m}^3$, V is the average speed of MagSTARS, 6.94 km/s, and A is the area of MagSTARS' ram face, 0.0285m^2 , as described in Section 3.1. The generalized calculated value of the drag force experienced by MagSTARS is $6.77 \times 10^{-9} \text{N}$. This value is a general estimate for understanding the effects of atmospheric drag - a much higher fidelity drag model was used in orbital calculations, as described in Section 4.3.

Although this force seems small, it has to be considered over the mission lifetime because it will cause MagSTARS' orbit to decay. To counteract this, once the orbit decays past a certain

tolerance, stationkeeping maneuvers are performed to raise the satellite to the upper bounds of its original orbit, as described in Section 4.3. Astrogator, the STK tool used to model MagSTARS' orbit, uses a provided drag coefficient and cross-sectional area to calculate the drag on the satellite during its orbit. When the orbit is propagated with Astrogator, drag is accounted for and the orbit decays accordingly. Section 4.3 discusses the stationkeeping process in depth and includes mission lifetimes with and without stationkeeping maneuvers.

8.1.3.2 The Plasma Environment and Spacecraft Charging

Much of the ionosphere is made up of plasma, or a mixture of free electrons and positively charged ions. In LEO and particularly in polar orbits like MagSTARS', plasma causes an effect known as spacecraft charging. Spacecraft charging is defined as the buildup of electrical charge on the surface and interior of the spacecraft, and it can negatively affect sensitive onboard electronics or cause errors in instrumentation readings. Typically, spacecraft charging is not a high concern for LEO orbits, but spacecraft in polar LEO orbits experience a higher amount of charging because of auroral activity.

Surface charging occurs in two forms – absolute and differential charging. Absolute charging occurs when the entire spacecraft surface is given a net potential charge relative to the plasma around it. Because absolute charging is uniform, it does not pose much danger to the spacecraft, its electronics, or its instrumentation. Absolute charging occurs when the entire surface of the spacecraft is made of conducting materials, so the charge can distribute evenly over the surface.

Differential charging can be much more dangerous. Separate areas of the spacecraft can be charged to different electric potentials, as they are exposed to different environmental conditions,

and if some areas of the spacecraft surface are made of non-conductive materials, the charge will not distribute evenly over the spacecraft surface and a potential difference will be created. This potential difference across the spacecraft surface can lead to electrostatic discharges or surface arcing, which can damage equipment or affect data collection.

Internal charging occurs when electric charge builds up in the interior of a spacecraft due to penetrating electrons, and depends on shielding thickness, material conductivity, and the interior spacecraft geometry. If this buildup of charge causes dielectric materials inside the spacecraft become too charged and reach their breakdown strength, dielectric breakdown and arc discharges can occur, which can lead to major damage of crucial spacecraft systems [95].

Calculating the effects of spacecraft charging is a complex process that is outside of the scope of this project. However, this is an aspect of the mission that must be considered, and it is recommended to be a focus of research in future projects.

8.2 The Magnetic Field Environment

Earth's ambient geomagnetic field and those magnetic fields induced by onboard current sources present another component of the environment that must be considered. Varying or high magnetic field intensity can cause errors in equipment readings or damage onboard computers. The hardware on MagSTARS includes instruments which require magnetic field strength to be within a certain range of intensities to function correctly. For example, a magnetometer is to be used for the ADC subsystem and its functionality requires knowledge of the Earth's unperturbed magnetic field. The magnetorquers induce their own magnetic field, thus the interaction between Earth's magnetic field and this internally generated magnetic field must be modelled to determine if any magnetic shielding is required for specific MagSTARS instruments at their location. First,

analysis of the ambient magnetic field during the MagSTARS mission was performed and second, analysis was done on the induced magnetic field by the magnetorquers.

8.2.1 Analysis of the Earth's Magnetic Field

STK's Space Environments and Effects Tool (SEET) includes several magnetic field models, derived from the International Geomagnetic Reference Field (IGRF). The IGRF is a multi-pole spherical harmonic expansion that computes the strength and direction of Earth's magnetic field given geocentric coordinates. Because the Earth's magnetic field changes over time, the IGRF relies on coefficients calculated about once every 5 years, and STK extrapolates these coefficients for other times from this past data. The "Full-IGRF" model was selected as the best option for providing magnetic field data. The option for calculating the external magnetic field was turned off, as this is affected by solar weather and mainly has large effects at high altitudes [64].

A portion of the magnetic field data is shown in Figure 60. This graph shows the overall magnetic field intensity in nanotesla (nT) with respect to MagSTARS' orbit during one orbital period of the science leg on January 1, 2021. The magnetic field strength does not vary linearly with altitude, instead it peaks around 260 and 380 km at 52000 nT, with a minimum of about 23000 nT at 415 km. However, this graph does not describe the entirety of Earth's magnetic field, as it varies with location as well as altitude across the globe - it is simply a snapshot of some of the data available.

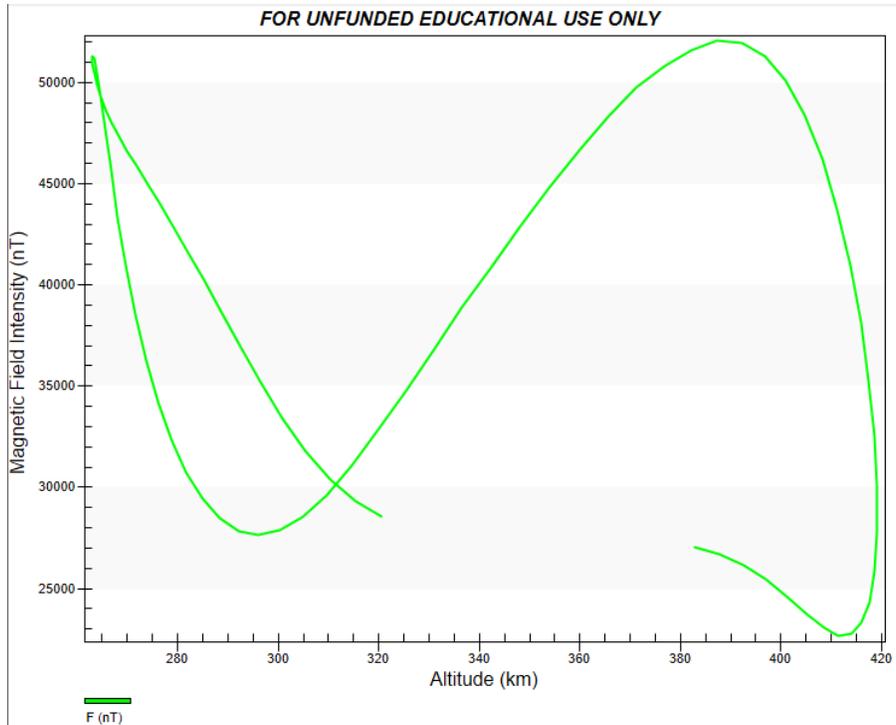


Figure 60: Magnetic field intensity with respect to altitude for one science leg orbit of MagSTARS from STK

Magnetic field analysis over the entire mission duration shows that MagSTARS will experience a maximum magnetic field intensity of 60840 nT and a minimum magnetic field intensity of 17670 nT with a mean intensity of 39190 nT.

8.2.2 Analysis of the Induced Magnetic Field by the Magnetorquers

In order to ensure that the magnetic field generated by MagSTARS’ magnetorquer did not interfere with the magnetic field readings by the onboard magnetometer, a magnetorquer and its generated magnetic field were modeled in COMSOL Multiphysics. Originally, this model was compared to a high-quality magnetometer used by the ADCS. However, it was discovered during the structural design process that the magnetorquer board included a magnetometer, so the external magnetometer was deemed unnecessary. This “included” magnetometer was near enough to the

magnetorquer to be affected by the generated magnetic field. It was assumed that the magnetorquer board included an induced counteractive magnetic field or code adjustments to compensate for this, however, the documentation included no mention of these counteractive measures. Because of this lack of documentation, it was decided to include a lower-quality external magnetometer as a failsafe option for the ADCS. This magnetometer was lighter and smaller, satisfying mechanical design needs, but also functional enough to provide backup should the “included” magnetometer be affected by the generated magnetic fields.

The magnetorquer was simulated in COMSOL as a cylinder that generated a magnetic field as an actual magnetorquer coil would, as seen in Figure 61. The magnetic flux density norm was calculated within a cylinder centered on the magnetorquer with a radius of 0.3 m and a height of 0.5 m, which encompasses MagSTARS’ entire structure.

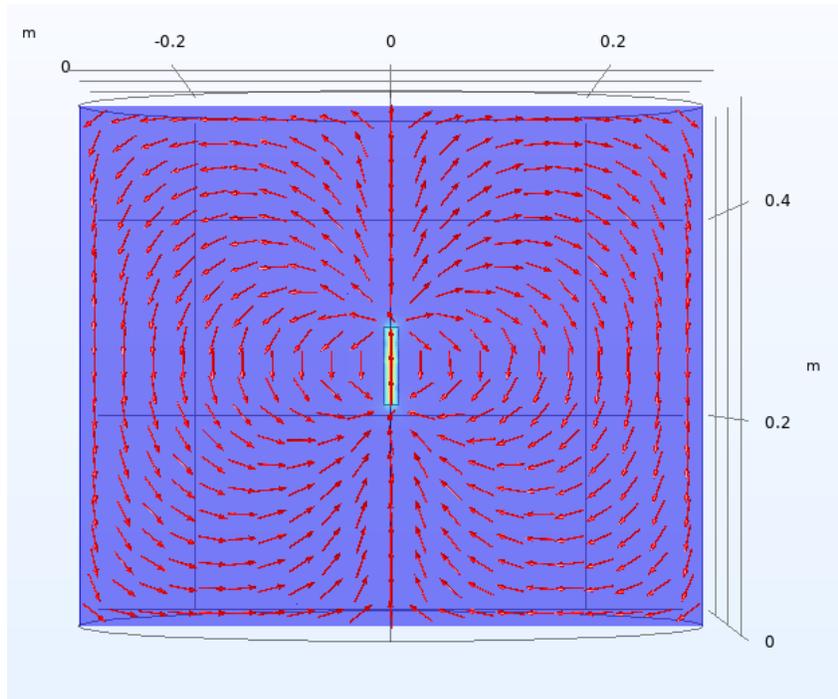


Figure 61: Magnetorquer and magnetic field modeled in COMSOL

The vector distance of the magnetometer on the magnetorquer board from the magnetorquer itself was measured using a CAD file provided by the developer. This vector was then included in the COMSOL model as a 3D cut line originating at the center of the magnetorquer, and the magnetic flux density norm, or B field, was calculated along this line. At the magnetometer's radial distance of 29.21 mm along the line, the magnetic field was approximately 1.38 G, which was enough to cause a bias in the magnetometer's reading of the external magnetic field. As mentioned earlier, there are most likely correction measures in place to compensate for this, but because of the lack of documentation it was decided to add an external magnetometer as a failsafe.

The distance between the external magnetometer and the magnetorquer was calculated using the MagSTARS structural model, and the magnetic field along this vector was calculated in the same manner as before. The external magnetometer was located 222 mm away from the magnetometer along this line. At this distance, the B field generated by the magnetorquer was .0159 G, which is small enough to not have a negative effect on readings of the external magnetic field. Therefore, no magnetic shielding between the two devices is required.

8.3 The Radiation Environment

Radiation is another environmental factor that was considered. Charged particles found within the magnetosphere make up Earth's naturally occurring, trapped radiation environment. As spacecraft spend more time in orbit interacting with this radiation environment, particles within the spacecraft structure are slowly ionized, which can damage electronics and equipment if the dose is over a certain radiation tolerance. The total amount of radiation imparted on the spacecraft over a set period of time is known as the Total Ionizing Dose (TID) [96]. Another aspect of the

radiation environment that was considered is the un-trapped radiation environment, which consists of single particle events such as solar energetic particles (SEPs) and galactic cosmic rays (GCRs). These are highly charged particles that enter Earth's magnetosphere from locations outside Earth's natural radiation environment. SEPs are particles that originate from the Sun and are common byproducts of solar flares or coronal mass ejections. GCRs are particles that reach Earth from outside the solar system, usually originating from large extrasolar events such as supernovas. The amount of GCRs that enter Earth's magnetosphere is affected by solar wind, with periods of low solar activity allowing the GCR particles easier access to Earth. Both SEPs and GCRs can cause single event upsets, in which an energized particle collides with the satellite and causes a glitch or upset in the electronics equipment [64]

8.3.1 Radiation Analysis: Total Ionizing Dose

In order to ensure that the total radiation dose received by MagSTARS will not cause equipment damage or failure, MagSTARS' TID over its mission lifetime was calculated and compared with the radiation tolerances provided for each component. MagSTARS does not have an exceptionally long mission life, so the TID is not a large concern as it is a function of time, but it was still important to calculate and to ensure the correct precautions are taken, if required.

The radiation tolerances for each component of MagSTARS can be seen in Table 30. Tolerances were taken from component datasheets, or if this information was not available for a specific component, the tolerance for a similar component was used. In the few cases where no radiation tolerance data was available for similar components, a tolerance of 10 krad was assumed, as this was the lowest researched tolerance for the other components.

Table 30: Radiation Tolerances for All MagSTARS Components

Component	Radiation tolerance (krad)
NanoAvionics MTQ3X Magnetorquer	20
Hyperion GNSS200 GPS	> 36
NSS NFSS-411 Sun Sensor	20
NSS NCSS-SA05 Sun Sensor	20
Spacemanic SM-ADS-AG4 Magnetometer	10
Control Board ISIS On Board Computer	>15
BEI Model 4310 Linear Accelerometer	10
ARS-15 MHD Angular Rate Sensor	10
SLink-PHY S Band Transceiver	10
Spacecom Patch Antenna	10
Mini-INMS (science payload)	10
ISISpace Modular Electric Power System	20
ISISpace Solar Panels	>2 years in LEO
BET-1mN (with double tank)	>10

STK’s SEET tool has several radiation computational modes that can be used to predict the total ionizing dose for different levels of radiation shielding. It was determined that the default mode, *Radiation Only* was a sufficient model to perform calculations, as it accurately models local Earth-atmosphere radiation for low altitudes. *Radiation Only* computes radiation dose rates and integrated doses using data from the APEX Space Radiation Dosimeter and the Combined Release and Radiation Effects Satellite [64]. This model was applied to the full-mission MagSTARS STK

scenario and TID data was gathered in a report. This computational mode only calculates TID for a satellite with 82.5 Mils, 232.5 Mils, and 457.5 Mils of aluminum shielding. MagSTARS will have an aluminum exterior of 145 Mils, so the actual TID was extrapolated from the 82.5 Mils and 232.5 Mils values. These STK-determined values as well as the extrapolated final TID can be seen in Table 31.

Table 31: Radiation Dosage Over the MagSTARS Mission

Shielding Thickness (Aluminum)	82.5 Mils	232.5 Mils	145 Mils (actual)
TID	46.95 rad	31.56 rad	40.54 rad

The total ionizing radiation dose for MagSTARS’ lifetime was calculated to be 40.54 rad, which was over an order of magnitude below the lowest component tolerance of 10 krad. The solar panels tolerance data is provided in years, and because MagSTARS’ mission life is approximately 180 days, that tolerance was met as well. Therefore, it was concluded that because the TID was within all component tolerances, no supplemental radiation shielding was required.

8.3.2 Radiation Analysis: Solar Energetic Particles

The amount of SEPs that a satellite will encounter is dependent on solar activity. High levels of solar activity spawns high amounts of SEPS, and vice versa. Although precise amounts of solar activity cannot be perfectly predicted, the Sun does follow an 11-year solar cycle. Approximately every 11 years, the Sun’s magnetic poles flip, causing periods of elevated solar activity. Solar activity reached a minimum in late 2019, marking the beginning of Solar Cycle 25,

with an expected solar maximum in mid-2025 [97]. The MagSTARS mission was modeled from October 2020 to July 2021, a period of mid-level solar activity.

STK's SEET tool has several SEP computational models based on different datasets that can be used to calculate the probabilities of fluence levels of different energy SEPs. The Rosenqvist et al. model was chosen as the most applicable to the mission. This model includes expected and predicted SEP fluence based on observational data from the Interplanetary Monitoring Platform satellites between 1963 and 1991, as well as supplemental observational data from the GOES satellites between 1974 and 2002. Predictions of proton fluence in this model are made based on the probability of large solar events occurring around solar maximum years, which is derived from the aforementioned data [64]. Figure 62 shows the probability of MagSTARS receiving different fluences of several set particle energy levels, generated with STK's Rosenqvist model over MagSTARS' lifetime.

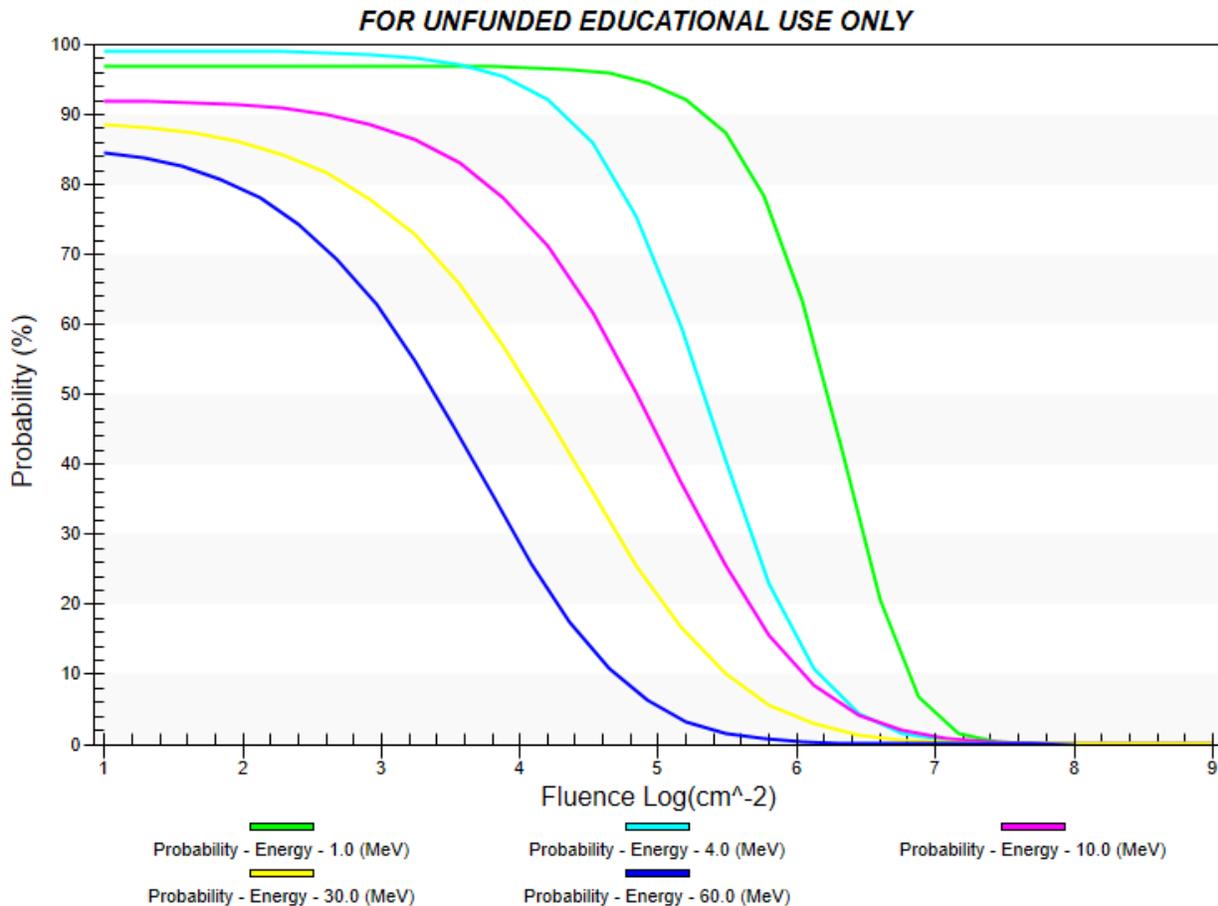


Figure 62: Probability of SEP fluences on MagSTARS predicted by STK

Figure 62 shows very high probabilities of low SEP fluence for all energy levels, which is corroborated by this being a period of mid-range solar activity and MagSTARS having a short mission life. Furthermore, the highest energy level fluences have the lowest probability. Because particles with higher energy levels are more likely to cause single event upsets, this means the likelihood of these events occurring is low. The low probability that MagSTARS will be subject to large amounts of SEPs, especially the high high-energy ones, means that SEPs are of low concern. In general, increased shielding would offer further protection from damage or errors caused by SEPs, however low probability these events may be. However, this shielding would

increase both the weight and cost of MagSTARS, and as the probability of damaging events caused by SEPs is low, it was determined shielding was not required.

8.3.3 Radiation Analysis: Galactic Cosmic Rays

STK's SEET tool also has several models for calculating the GCR flux. The Badhwar O'Neill 2010 model, or BO10 model, was selected because it included the most up-to-date data and could be applied to all dates, unlike another model that could not be used for dates past 2015. The BO10 model extrapolates GCR fluxes from GCR flux data collected from 1955 to 2010 by the NASA Advanced Composition Explorer Cosmic Ray Isotope Spectrometer and supplements these extrapolations with solar activity data. Figure 63 shows the differential GCR flux for different particle energy levels over the mission lifetime, calculated by STK with solar maximum conditions [64].

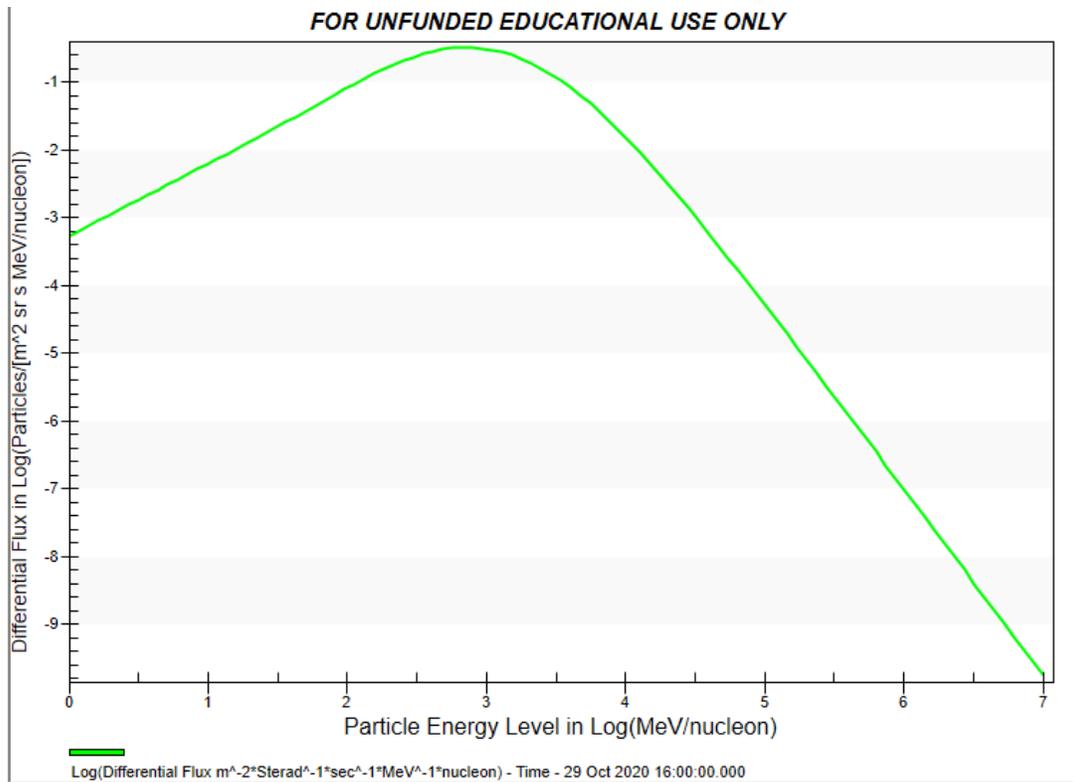


Figure 63: GCR flux vs. particle energy level over MagSTARS' mission duration from STK

The highest flux occurs from the lower energy level particles, which is beneficial – lower energy particles have less of a chance of causing detrimental electronics damage. The highest flux level, approximately $-0.5 \log(\text{Particles}/(\text{m}^2 \text{ MeV}/\text{nucleon}))$, occurs for particles with an energy level of approximately $3 \log(\text{MeV}/\text{nucleon})$. Both this energy level and flux are still low when compared to known damaging amounts. This result is further supported by the knowledge that GCRs are commonly not a cause for concern for low Earth orbits. Earth's magnetosphere and upper atmosphere block higher energy GCRs, and while some do pass these barriers and can even be detected on Earth's surface, they are usually low energy and not damaging. In most spacecraft designs, mitigating measures, such as shielding, against GCRs are not considered unless the spacecraft is in a geosynchronous orbit or higher [64].

8.4 Particle Impacts

Earth's orbit is filled with debris, both human-made and naturally occurring. Small particles of space debris or particulate meteors constantly collide with spacecraft, though most impacts are too small to cause damage. However, larger particles moving at high velocities can damage the mechanical components of a spacecraft, causing loss of efficiency, errors, or even mission failure. Therefore, it was important to model these impacts and understand the effect they will have on the mission and what shielding, if any, is required.

8.4.1 The Meteor Environment

STK's SEET tool has models for computing both meteor and particulate human-made debris over the mission lifetime. SEET's meteor model is based on a database of 50 parameterized meteor showers. These meteor showers occur as Earth passes through the dust trails left behind by comets that have passed by on their own orbits and can usually be predicted as these times are well documented. They usually consist of particles with masses ranging from 1 μg to 10 mg and velocities from 12 km/s to 70 km/s. Generally, documented showers are weak with a low Zenithal Hourly Rate, or the number of observable shower meteors per hour. However, higher Zenithal Hourly Rate showers, known as meteor storms, do occur [64].

The database of showers that STK's meteor environment utilizes, the Jenniskens database, consists of 10 years of documented visual shower observations that are translated into mass-dependent flux rates. This database is supplemented with a particle mass distribution profile that created background flux from sporadic meteors, or cosmic dust not associated with a particular shower event [64].

A cross-sectional area of 0.028538 m², MagSTARS’ ram face area, was used for the meteor calculations. The model also requires a “pit depth” to be specified, or the largest depth of particle impact the satellite can handle before something is damaged. A pit depth tolerance of 0.5 mm was used, as this is approximately one-sixth of the thickness of MagSTARS’ exterior and determined to be a sufficient factor of safety. The results of the STK meteor environment model applied to MagSTARS’ lifetime can be seen in Figure 64.

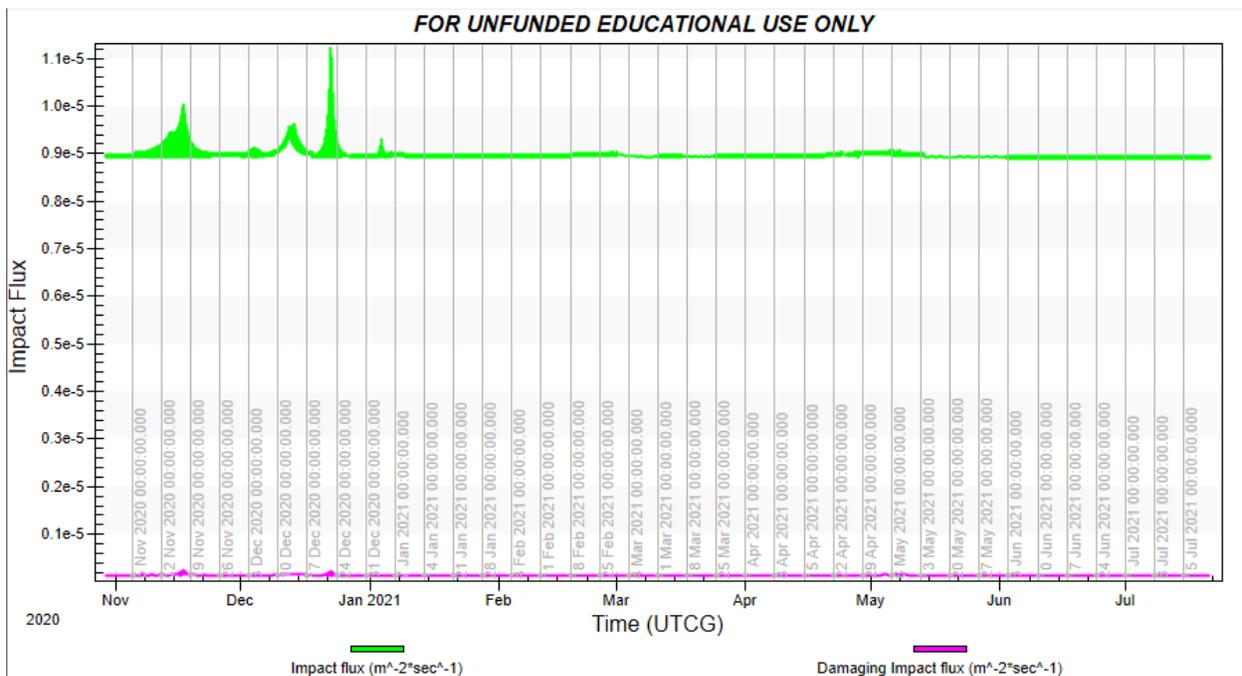


Figure 64: Particulate meteor impact flux over MagSTARS’ mission duration from STK

The green line in Figure 64 is the rate or flux of particle impacts across the mission lifetime. The pink line represents the impact flux that is predicted to have a damaging effect on the satellite. It appears that the first few months of the mission are during meteor showers based on the high flux spikes through November and December, and that the rest of the mission is only subject to background sporadic meteor flux. The damaging flux only climbs slightly above zero twice, once in mid-November and once in late December. However, these fluxes are still so small that the risk

of this slight damage occurring was determined to be acceptable, as increasing shielding in this case meant using thicker aluminum, which would increase MagSTARS' weight and therefore have a negative effect on propulsion and all the subsystems related to it. Based on the problems increasing this shielding would cause compared to the very slight risk meteor impacts pose, it was decided not to add additional shielding to protect against this environmental effect.

8.4.2 The Debris Environment

Human-made particulate orbital debris impacts were calculated using STK SEET as well. Orbital debris takes many forms, from large objects such as abandoned satellites and ejected rocket stages that can be tracked from the ground to small particles that have broken off of these large objects as a result of collisions, erosion, or fragmentation. The SEET Debris Environment calculated a likely number of collisions between these particles and MagSTARS using Kessler's empirically derived equations for calculating orbital debris particle flux. These equations are based on long periods of observation of real orbital debris flux, and are a function of particle size, average satellite altitude, orbital inclination, and solar activity. An average impact velocity profile for the debris is modeled as a function of orbital inclination, and the average increase of space debris over time is predicted using a fixed annual net growth rate. The averaged orbital and environmental parameters are used to calculate yearly-averaged (or, if the mission duration is less than one year, mission-averaged) particulate debris flux values [98]. This data is output as an average rather than specific values for each time step, as the calculations are based on long-term averages of empirical data and it would not be appropriate to translate this for short time intervals [99]. Using a user-specified damaging pit depth, several algorithms by McDonnell and Sullivan are then used to determine the proportion of impacts that will cause damage to the spacecraft [64].

Similar to the Meteor Environment, a pit depth of .5 mm and a cross sectional area of 0.028538 m² was specified for the Debris Environment calculations. However, the Debris Environment is not available for dates after December 30, 2020. The MagSTARS orbit was modeled in STK from October 2020 to May 2021, so the scenario dates had to be pushed back a year, starting in October 2019, to complete these calculations. This meant the orbital debris data was calculated under slightly different conditions than the rest of the mission analysis, however the results should be similar. Table 32 shows the average debris impact fluxes for the MagSTARS mission calculated with STK.

Table 32: Mission Averages of Debris Impact Flux

Impact Flux (m ⁻² s ⁻¹)	Mass impact flux (kg/(m ² s))	Damaging Impact Flux	Damaging mass impact flux
$5.46 \cdot 10^{-6} \text{ m}^{-2}\text{s}^{-1}$	$2.688 \cdot 10^{-16} \text{ kg/m}^2\text{s}$	$1.070 \cdot 10^{-8} \text{ m}^{-2}\text{s}^{-1}$	$2.615 \cdot 10^{-16} \text{ kg/m}^2\text{s}$

Approximately 0.2% of the total average debris impact flux was identified as damaging. These damaging impacts accounted for approximately 97% of the mass flux, which aligns with the knowledge that larger particles have the potential to cause more damage to the spacecraft structure. However, the mass flux of these particles was still very low, and “damaging” in this case simply means penetration through one-sixth of MagSTARS’ structure – these particles will not enter MagSTARS’ interior. The low mass flux, coupled with the low percentage of damaging impacts, meant the same conclusion was drawn as with the Meteor Environment: the cost and weight of adding extra shielding outweighs the small probability of a particle impact that would negatively affect the mission.

9 Thermal Control Analysis

This section consists of the theory, analyses, methods, and results of the processes undertaken to complete the objectives of the thermal control subsystem of MagSTARS.

9.1 Thermal Overview

Outside of the protective layers of Earth's atmosphere, exposure to solar radiation is far more dangerous. This energy is brought on by periods of illumination and is referred to as solar flux. The Sun's solar flux at Earth is 1367 W/m^2 on average and ranges from about 1317 W/m^2 to 1419 W/m^2 . This range in the solar flux is caused by the variance in the distance between the Sun and the Earth over the course of a year [100]. In addition to the transfer of thermal energy from the Sun, two other forms of thermal radiation affect spacecraft in LEO: the reflection of Solar radiation from Earth, known as "albedo", and the blackbody radiation that is produced by Earth [101]. The albedo fraction of Earth ranges from approximately 0.18 to 0.55 with 0 meaning that all of the solar energy hitting Earth's surface is absorbed and 1 meaning that all of the energy is reflected back into space [42], [100]. The blackbody radiation of the Earth is much smaller than the Sun's, at around 231 W/m^2 on average. Additionally, the heat produced from power dissipation within MagSTARS needed to be considered. This heat caused by inefficiencies in power-consuming devices is important to quantify, as it could radiate quickly throughout the spacecraft due to the proximity of the components on MagSTARS [100].

All these various sources of thermal energy were accounted for in order to accurately model the temperature MagSTARS would experience. This was done to ensure that the MagSTARS did not exceed the temperature ranges of the components onboard. Each component on MagSTARS had its own operating and survival temperature ranges. The operating temperature range is the

interval over which the device can operate at full capacity, and the survival temperature range is the interval over which the device will not accumulate long term damage [100]. The temperature of the CubeSat must not go outside the survival temperature range at any point of the mission, whereas the operating temperature only needs to be maintained for the phase when a component is operating.

While many of these devices have similar thermal limits, some components have a smaller thermal range and thus are more susceptible to temperature changes [48]. For example, as stated in Section 5, batteries are found to be particularly sensitive to cold temperatures [61]. Relatively cold temperatures cause the depth of discharge of a battery to increase, which makes it lose energy much faster than it would at temperatures within the range of its operating temperature. To ensure that batteries do not fall below this temperature range, most batteries designed for spaceflight are manufactured with a built-in heater, a method of thermal control [42].

To combat this harsh environment and guarantee the survival of the devices on board MagSTARS, several methods of thermal control had to be considered. These methods were defined as either passive or active. Passive methods are means of decreasing thermal energy without using energy from the spacecraft in the process. Some examples of passive methods are radiators, heat pumps and surface finishing. Radiators release thermal energy within the spacecraft via exterior panels with high emissivity. Heat pumps use water to move heat from the interior to the exterior of the spacecraft by allowing convection currents to move the water. Surface finishing makes the spacecraft more reflective, allowing the spacecraft to emit more thermal energy from external radiation [61]. This can be done by either applying a lighter color of paint to the exterior, or by making the exterior of the satellite out of a more emissive material. Figure 65 shows the absorbances and emittances of various materials.

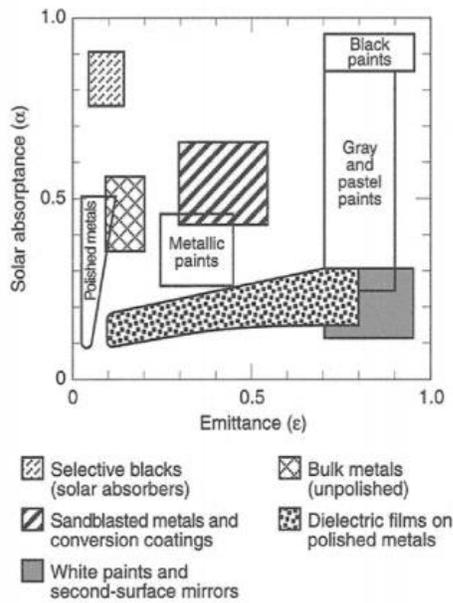


Figure 65: Absorbance and emittance of various materials [90]

Contrasting a passive method, an active method is one that uses a device, that requires energy, to regulate thermal energy and is usually used when passive methods are not enough to maintain an adequate temperature. The heaters previously mentioned to ensure batteries remain operational are an example of an active method of thermal control. Two other methods are thermoelectric coolers that remove heat by running current through two different semiconductors, and a louver which is a mechanical shutter that is used to release thermal energy if the spacecraft's interior becomes too hot [100], [42]. Figure 66 displays the design of a louver.

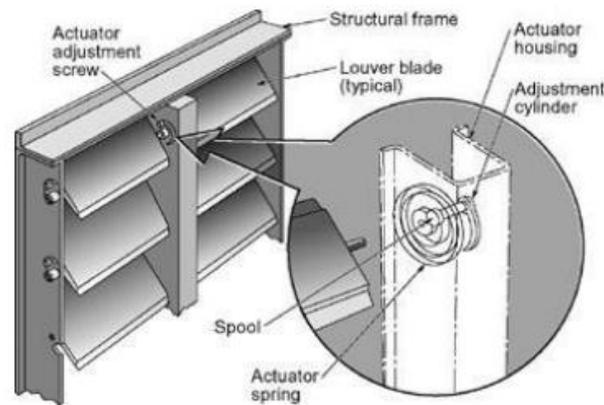


Figure 66: Design of a louver, an example of active control [42]

Passive methods are generally preferred as they require no additional power usage and are much simpler to employ when compared to active methods. For this reason, as well as the ease of changing the emissivity of a material in modeling software, using reflective surface finishing is one of the more widely used options for thermal control [61].

The thermal energy MagSTARS would experience on its mission was modeled in order to decide which methods of thermal control, if any, were required. The thermal load of the system over time due to external radiation was modeled in STK, and the values of solar intensity, nadir vector and Sun vector over time were obtained. The solar intensity is the measure of direct Sunlight impacting the spacecraft and is used to find solar flux. The Sun vector is the directional value of the solar intensity. The nadir vector is the vector between the spacecraft and Earth and is used to find blackbody radiation, as it shows what part of the spacecraft is facing the Earth throughout the flight [48].

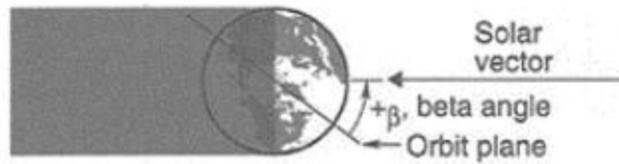


Figure 67: Beta angle in relation to the Sun vector [90]

The COMSOL Multiphysics modeling software was used to create a specific type of model called a “Heat Transfer with Surface-to-Surface Radiation” model, which displayed the heat from internal radiation within MagSTARS in addition to the heat from external radiation. The CAD model used in this analysis was a greatly simplified and defeatured SolidWorks model imported into COMSOL [42]. The complexity of the model was reduced due to the difficulty and time-consuming nature of modeling heat transfer for a highly complex model. Smaller components, such as fasteners and wiring were removed from the model due to their minimal influence on the overall thermal behavior. Larger elements, such as batteries and sensors, remained in the model but with greatly simplified geometry. Previous MQPs had assumed the average values of Earth’s albedo and the solar flux were sufficient for accurate analysis, however it was recommended in the 2017 MQP report *Design and Analysis of the Sphinx-NG CubeSat* to increase the accuracy of the thermal analyses by creating two different simulations in which the albedo and solar flux were at their greatest and smallest possible values [42]. Thus, due to this advice supported by the reasoning of extreme ranges of values experienced, two different models were analyzed: a hot case and a cold case. The two cases modeled the warmest and coldest possible temperatures, respectively, that MagSTARS may have endured [100], [42]. With MagSTARS accurately modeled for both of its temperature extremes, the temperature throughout the flight could be found and the appropriate thermal control method could be chosen.

9.2 Thermal Analysis Methods

The first step in thermal control of MagSTARS was to determine the lower and upper bounds of the temperature range the spacecraft could experience and remain within the operational temperature range during its mission. Using the data sheets of all the components on board MagSTARS, the operating temperatures of these components were found, as shown in Table 21. The tabulated components did not have data available on their survival temperatures, so it was assumed their temperatures range from -40 to 85 degrees Celsius, as this is the standard range for industrial grade electronics [102].

Table 33: Thermal Limits of Selected Components

Component	Operating Temperature [°C]	Survival temperature [°C]
Magnetorquers	-40 to 85	-40 to 85
GPS	-40 to 85	-40 to 85
Sun Sensor	-25 to 70	-40 to 85
Control Board	-25 to 65	-40 to 85
Accelerometer	-40 to 93	-40 to 85
Angular Velocity Sensor	-35 to 60	-40 to 85
Transceiver	-20 to 50	-40 to 85
Antenna	-10 to 45	-40 to 85
Science Payload	-10 to 50	-40 to 85
Battery	-20 to 70	-40 to 85

Many of the devices had an operational temperature within 20 degrees of the industry standard survivable temperature. Exceptions were found in the antenna, science payload, and

transceiver. The transceiver demonstrates an upper limit that was lower than the majority of other components used. The antenna and the science payload demonstrated a much narrower thermal range than most of the other components and thusly extra attention was paid to those devices as they were most susceptible to temperature changes.

The severity of exceeding operational temperature was determined by whether the device was being used at the time in which it was exceeded as well as the duration of time in which said thermal limit was exceeded. For example, if the thermal limit of the science payload was violated the mission plan would be consulted to see if the payload needed to be operating at that time, if it did not, then no thermal control was needed as long as the component was within survivable limits. If the science payload needed to be operational at that time the duration of time in which the limit was violated would be determined. If the time frame was small, such as on the order of minutes, the thermal control undertaken would be minimal.

The next step in analyzing the temperature of MagSTARS was to determine the heat generated by the satellite itself. Using data gathered by the power control subsystem certain components within the craft were determined to have the greatest power usage which consequently will have the highest heat output of the devices on board. Most of the components were found by the power subsystem to use less than one watt with outliers being the NanoAvionics Magnetorquers MTQ3X, the mini-INMS science payload and the propulsion system each having a maximum power usage of 1.2 W, 1.8 W and 15 at maximum performance, respectively. The total dissipated power of MagSTARS was then determined using (24).

$$P_{dissipated} = NP_{used} \quad (24)$$

The N in (24) is the percentage of inefficiency in each device. The components were assumed to have an efficiency of 90% in the same vein of a previous MQP [48]. The lack of information on efficiency from the manufacturers facilitated the necessity of these assumptions. The values for each of the components were taken from their maximum power usage during the mission.

Table 34: Power Dissipation of Selected Components

Component	Science Phase Dissipated Power (W)	Propulsion Phase Dissipated Power (W)
Mag-Torquers	0.12	0.12
GPS	0.0165	0.0165
Sun Sensor Fine	0.013	0.013
Sun Sensors Rough	0.005	0.005
Magnetometer	0.0010395	0.0010395
Control Board	0.04	0.04
Accelerometer	0.015	0.015
Gyroscope	0.03	0.03
Mini-INMS	0.18	0
IPDU	0.0066	0.0066
IPCU	0.0066	0.0066
IPBU	0.0063	0.0063
Antenna	0.01	0.01
Transceiver	0.05	0.05
Propulsion	0	1.5
Total Power	0.579	1.899

After accounting for the internal radiation acting on the spacecraft, the heat generated by external radiation was calculated. The STK scenario developed by the propulsion subsection was used to model MagSTARS and the Space Environments and Effects Toolkit (SEET) was used to

add in data needed to model the thermal elements of the scenario. SEET within STK was used to input the data for albedo, emissivity, absorptivity, the area of the side of the satellite facing the Sun and the total internal dissipation found earlier. The initial emissivity was entered as 0.85 and the initial absorptivity was entered as 0.92. These values were representative of the primary material in the Sun facing solar panel [42]. The total dissipation of MagSTARS was input as 0.579 W and 1.899 W corresponding to the total dissipation stated previously for the Science and Propulsion phases which act as the cold and the hot case respectively. The model in SEET was represented by a plate whose area was entered as 0.06 m^2 , corresponding to the area of the Sun facing side of MagSTARS which was 30 x 20 cm. This plate was modeled as being normal to the y-axis of the body of MagSTARS as to simulate the solar panel facing the general direction of the Sun over the course of the mission. Within STK the SEET temperature over the course of 24 hours were graphed.



Figure 68: MagSTARS STK model used for SEET thermal analysis

Next, solar intensity and the Sun vector reports were generated in STK. The solar intensity is a percentage between 0 and 100 that represents the extent of MagSTARS' exposure to Sunlight. The Sun vector, as previously stated, is the vector between the satellite and the Sun at any given time. The scenario was then changed to be representative of the cold case as the lowest possible values of albedo were entered into the simulation and the values of the solar intensity and Sun vector were recorded once again. The 3D graphics of the satellite displaying the vectors of the body of MagSTARS as well as the Sun vector are shown in Figure 68. The 3D graphics in STK used the default model of a satellite and not a custom model to represent MagSTARS, however the displayed model does not affect the analysis.

The data from STK was exported as a Microsoft Excel file with its time defined in terms of date. The files for the solar intensity and Sun vector were exported as .csv files as this file type is compatible with MATLAB. The timeframe was also converted into seconds rather than a date. This was done using a MATLAB code that was previously developed by a graduate student, Harrison Hertlein [48]. This code can be found in Appendix K. The solar intensity data was also divided by a factor of 100 in order to normalize the percentage. In addition to this the Sun vector file was separated into different Excel workbook files each representing the various directions of the Sun vector data, negative x direction, positive y direction etc. This was done to ensure that each side of the structure received the values of the Sun vector corresponding to its position. After this data was converted it was able to be imported into COMSOL Multiphysics.

COMSOL Multiphysics allows for the simulation of physical effects on a 3D model. COMSOL was chosen due to its accessibility and its compatibility with both STK and SolidWorks. COMSOL was also chosen as the previous CubeSat MQPs of 2017, 2018 and 2020 all used the program [42], [101], [48]. The data was imported into COMSOL as an Interpolation function,

which is when a function is defined by discrete points. This means that if the value of the function was not specifically defined at one time COMSOL will interpolate to find the approximate value at that moment [42]. The COMSOL model used the geometry of the CubeSat developed by the structural subsystem with it being greatly defeatured to allow for shorter computation times. The SolidWorks model of MagSTARS was saved as a .STEP file to prevent changes from the difference in software [42]. The COMSOL model was defined as a Heat Transfer with Surface-to-Surface Radiation model in order to accurately model the radiation of heat between different components. The body of the model was defined as an aluminum alloy Aluminum 7075-T6 and the solar panels were defined as Gallium Arsenide (Ga-As). The ambient temperature of the model was set to -270.45 °C, the temperature of space [101].

The geometry of the model was also defined as a form assembly rather than a form union in order to more accurately model MagSTARS. Within this form assembly mode, the boundaries of each side were treated individually meaning that the heat would not transfer unless in contact with each other, which accurately modelled heat transfer in space. In order to model the radiation between the various sides of the CubeSat the option to form an Identity pair between the sides was selected when changing the model from a form union to form assembly. The mesh of these models was created using a physics-controlled mesh with the Coarser setting selected.

A heat flux was modeled for each side of MagSTARS and was defined by the equation for solar flux (25). In (25), S is the solar constant represented with units of W/m^2 , SI was the solar intensity normalized and SV was the Sun vector.

$$Q_0 = S \cdot SI(t) \cdot SV(t) \quad (25)$$

Each radiation source was modeled onto MagSTARS using as an external radiation from the surface-to-surface heat transfer module. The value of solar radiation was modeled onto the side facing the Sun whereas the side facing the each was modeled with both the albedo flux and blackbody radiation of Earth. The sides were modeled to have a dissipation that equated to the material with a reference temperature of -270.45 °C representing the dissipation of the heat into

$$H = \alpha \times S \times F \times A \quad (26)$$

$$B = K \times A \times \left(\frac{Re^2}{Re^2 + h^2} \right) \quad (27)$$

space. The computations within COMSOL are directly dictated by equations for radiosity, irradiation and the blackbody emissive power.

$$J = \varepsilon e_b(T) + \rho_d \quad (28)$$

$$\varepsilon + \rho_d = 1 \quad (29)$$

$$G = G_m(J) + G_{amb} + G_{ext} \quad (30)$$

$$G_{amb} = F_{amb} e_b(T_{amb}) \quad (31)$$

$$G_{ext} = q_s + I_{diff} \quad (32)$$

$$e_b(T) = n^2 \sigma T^4 \quad (33)$$

$$q_{r,net} = \varepsilon(G - e_b(T)) \quad (34)$$

J represents the radiosity, or intensity of the radiation. ε represented the emissivity of the given material the calculation was being performed on and ρ_d representing the material's density. The Boltzmann constant is the physical constant related to the irradiation of a blackbody and is represented by σ . The irradiation was given as G . The irradiation as well as temperature due to the ambient conditions, in this case the temperature of space, were given as G_{amb} and T_{amb} . F_{amb} is the view factor for the ambient environment which in this case was one as it completely surrounds

MagSTARS. G_{ext} is the irradiation due to external radiation. I_{diff} is diffuse irradiance and was set to zero as there was none in this case. The parameter n is the transparent media refractive index, which was set to one as the body is opaque. q_s and $q_{r,net}$ represented the conductive heat flux and the net radiative heat flux. The blackbody hemispherical total emissive power is the power that is radiated off a blackbody object and is represented by e_b . G_m is the mutual surface irradiation, which measures the irradiation between two surfaces [103].

This COMSOL model was modeled over the course of 24 hours with a time step of an hour. Additionally, an entirely separate model of the interior components of MagSTARS was made due to the large computation times of such components being included in the model including external radiation. Each component was modeled as a heat source with the dissipation of that device modeled respectively. Once again this was modeled over the course of 24 hours with a one-hour time step. The setup for the COMSOL analysis of the internal dissipation is shown in Figure 69 and the setup for the analysis of external radiation is shown in Figure 70.

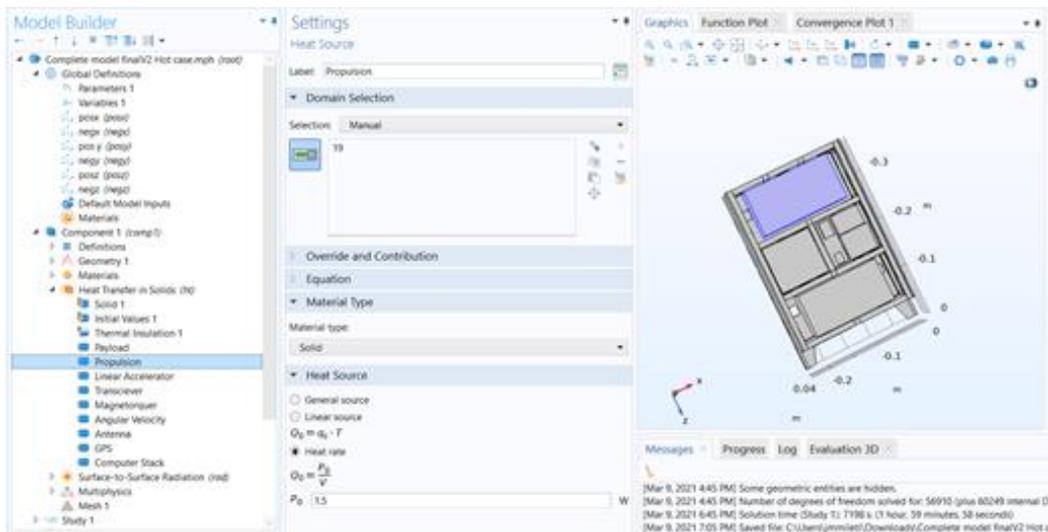


Figure 69: COMSOL model setup for internal dissipation

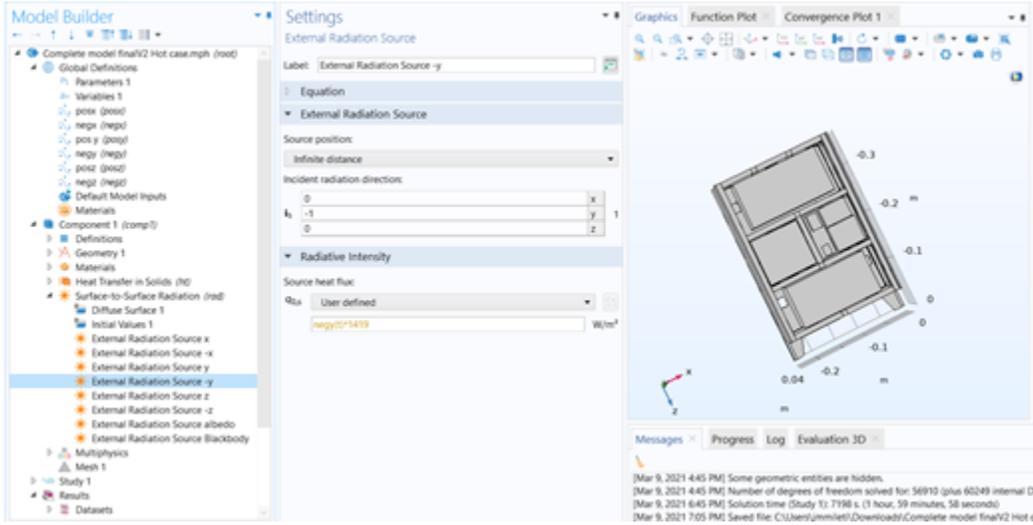


Figure 70: COMSOL model setup for external radiation

On the recommendation of a previous MQP, two different variations of the COMSOL model were made corresponding to the hottest and coldest possible cases for MagSTARS in order to ensure that the craft will remain within its limits in every possible scenario [42]. For each of these cases the solar constant and the albedo of the scenario were varied as show in Table 35 and Table 36.

Table 35: Hot Case Parameters

Parameter	Variable	Value [Units]
Solar constant	S	1419 [W/m ²]
Albedo constant	A	0.55

Table 36: Cold Case Parameters

Parameter	Variable	Value [Units]
Solar constant	S	1317 [W/m ²]
Albedo constant	A	0.18

The process of analyzing the external radiation of MagSTARS was repeated for these cases with the STK data and COMSOL models being changed. After these cases were modeled the temperatures experienced were compared to the survival and operational temperatures found earlier.

9.3 Results

In the STK simulation the period of illumination of MagSTARS is constant throughout the duration of the 24-hour simulation. The data produced by STK showed that MagSTARS will have a mean temperature of about 300 K with swings of about + or -70 K. For the cold case simulation, the temperature peaked at 366 K and reached as low as 235 K as shown in Figure 72. The hot case had very little difference in its temperature range with the upper limit being increased to around 373 K and the lower limit being increased to around 254 K as shown in Figure 71. In both of these cases the temperature briefly exceeded both the operating and survivability temperatures of the components onboard when the maximum and minimum temperatures were reached.

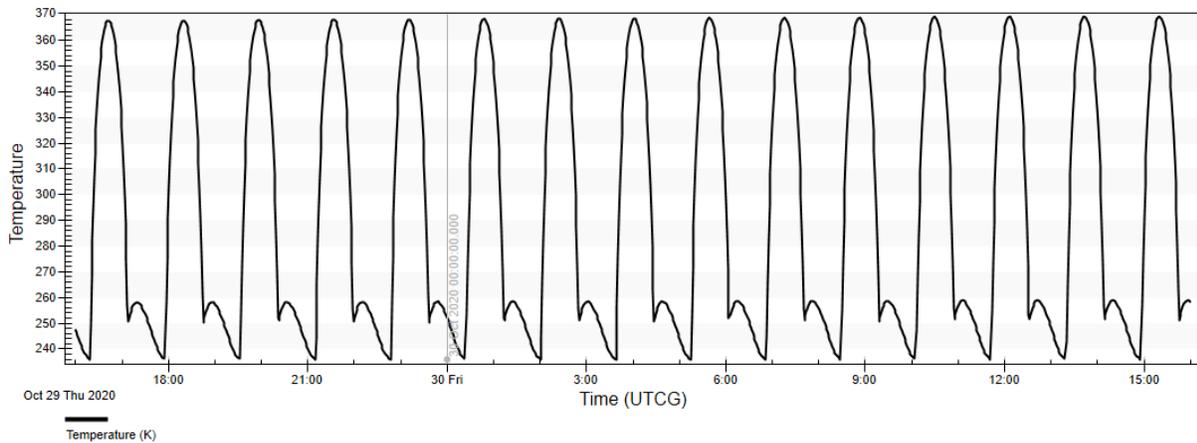


Figure 71: SEET temperature graph over 24 hours for the hot case

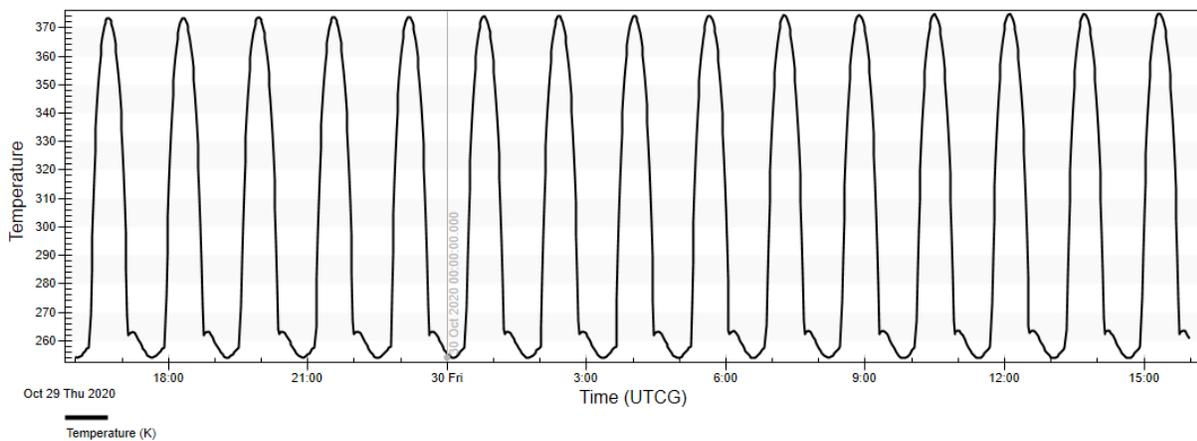


Figure 72: SEET temperature graph over 24 hours for the cold case

While the STK SEET tool provided a good estimate of the bulk temperature of the spacecraft from the solar flux, it did not account for internal heat sources, and did not accurately model dissipation or more complex surface conditions. Thus, these results were compared to the COMSOL model which offered higher fidelity in comparison to the SEET module. The values in Figure 71 and Figure 72 represented the average temperature of the satellite simulated on a very rudimentary object, whereas the COMSOL model more accurately represented the material of

MagSTARS as well as the positioning of the components on board. Thusly the COMSOL model was used as a more accurate gauge of the temperature.

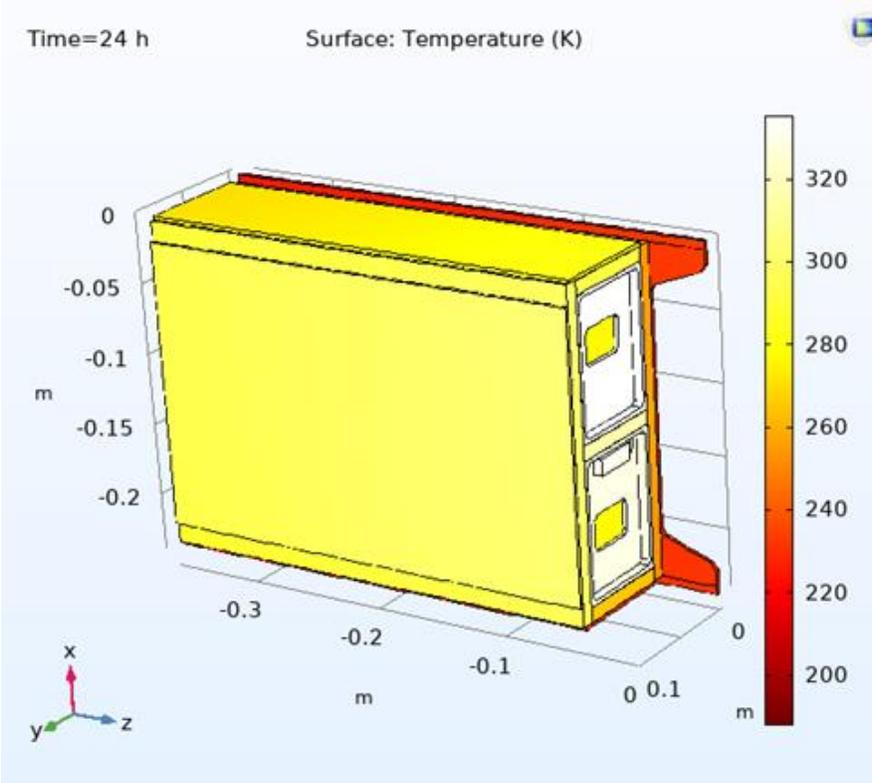


Figure 73: Sun facing exterior of the COMSOL hot case

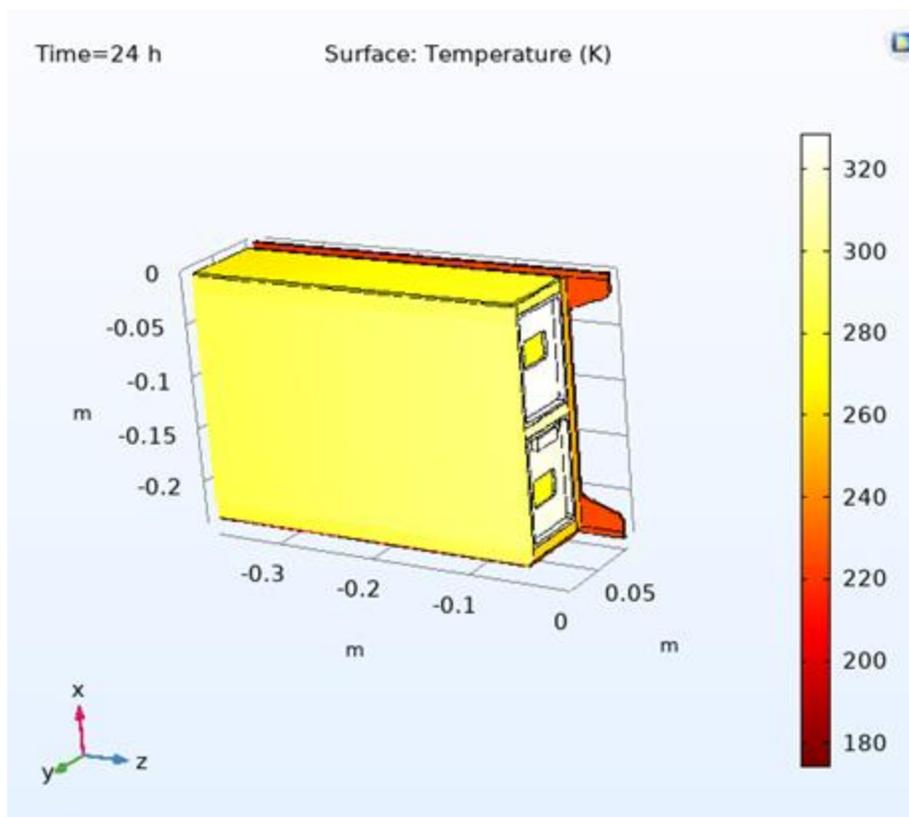


Figure 74: Sun facing exterior of the COMSOL cold case

The general temperature for the hot case displayed a range of 330 K to 190 K as shown in Figure 73 and the cold case displayed a range of 325 K to 180 K as shown in Figure 74. In both cases the solar panels were found to demonstrate the coldest and hottest relative temperatures depending on whether or not they were exposed to solar radiation. This is due to the composition of the panels differing from that of the frame. The COMSOL and STK model had similar ranges of values with the STK model being around 50 K hotter most likely due to the simulation being run on a highly simplified plate and not a more complex shape that would be more indicative of MagSTARS' actual structure and material make up.

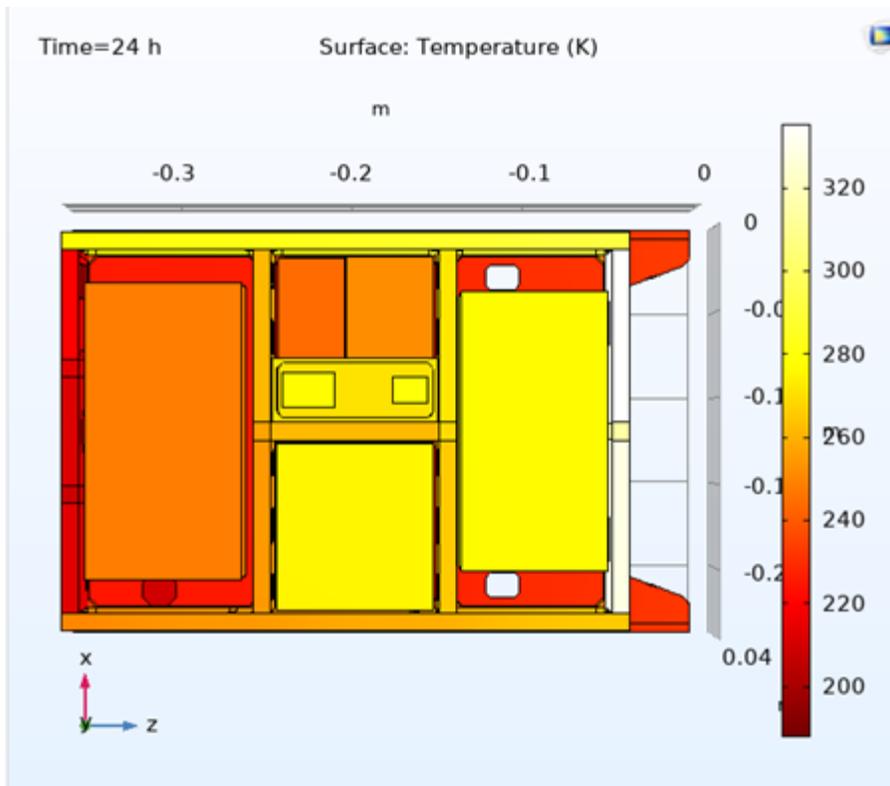


Figure 75: COMSOL internal components model for the hot case

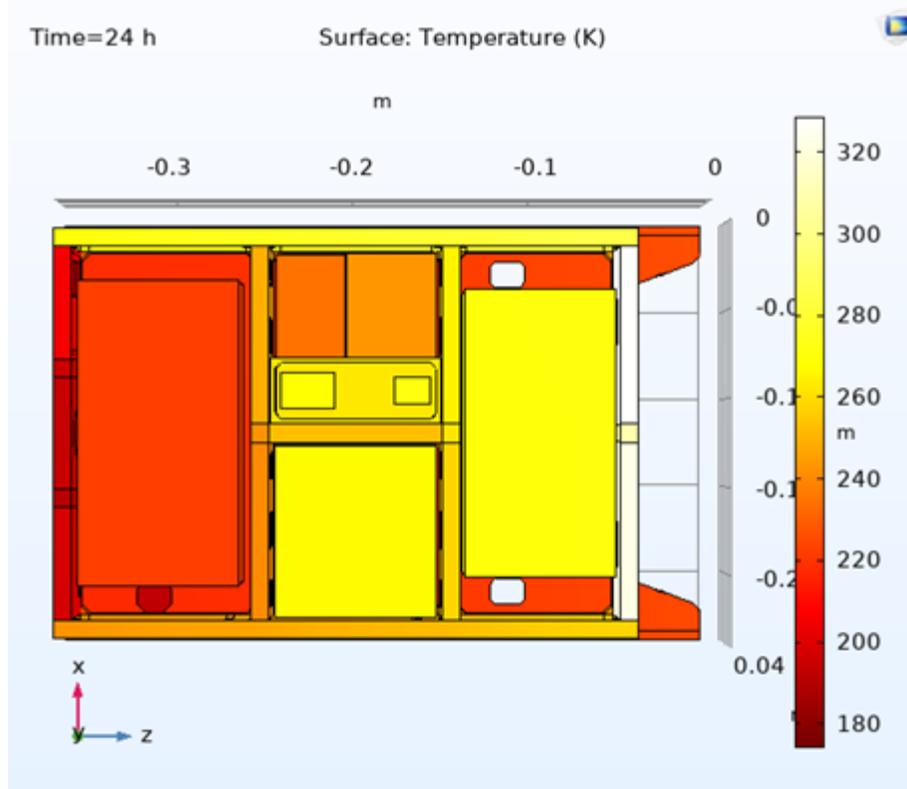


Figure 76: COMSOL internal components model for the cold case

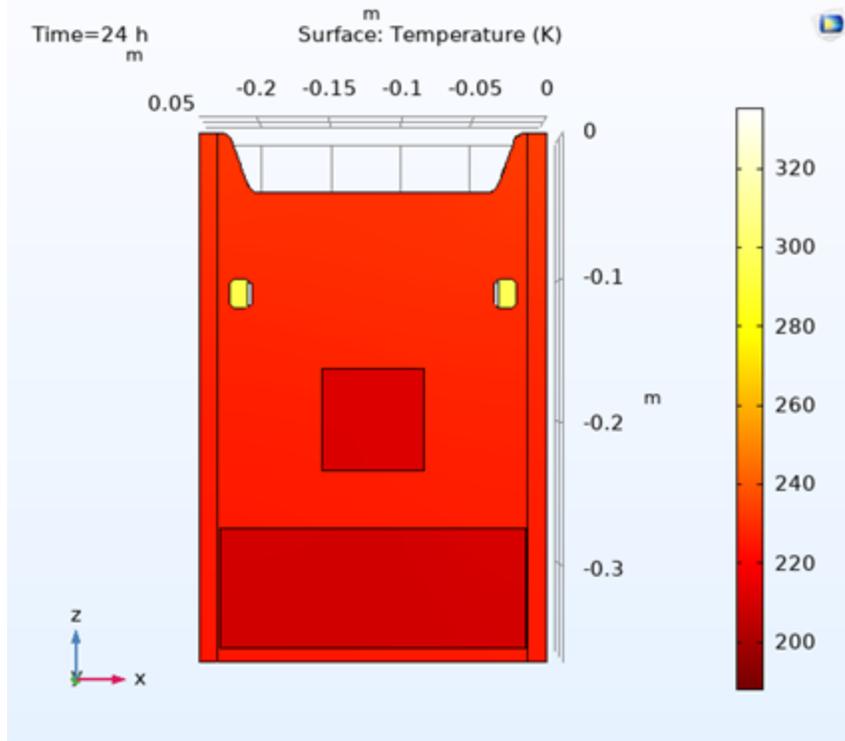


Figure 77: Side of MagSTARS opposite the Sun vector for the COMSOL hot case

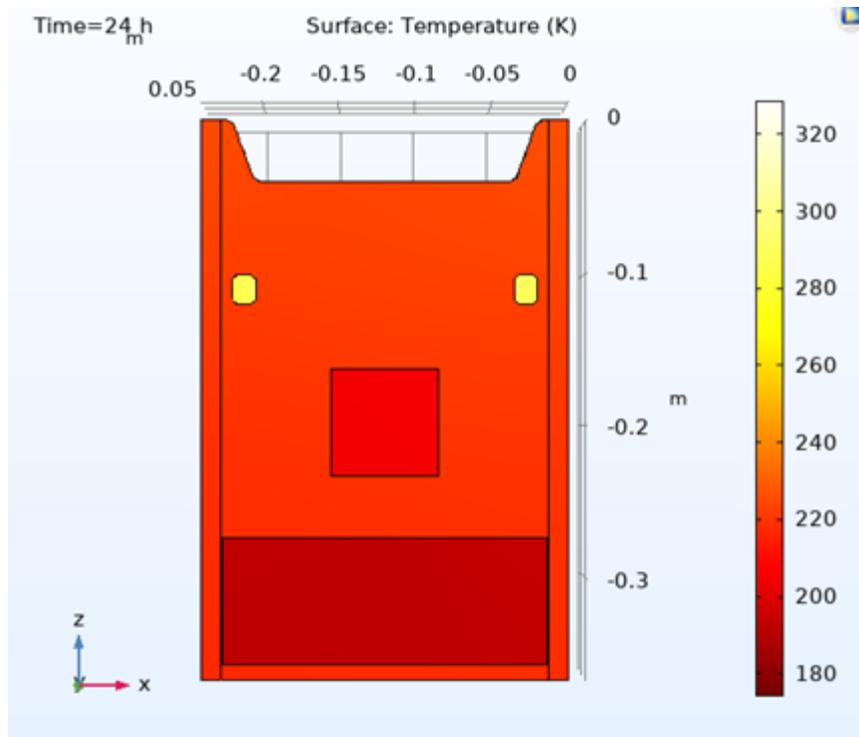


Figure 78: Side of MagSTARS opposite the Sun vector for the COMSOL cold case

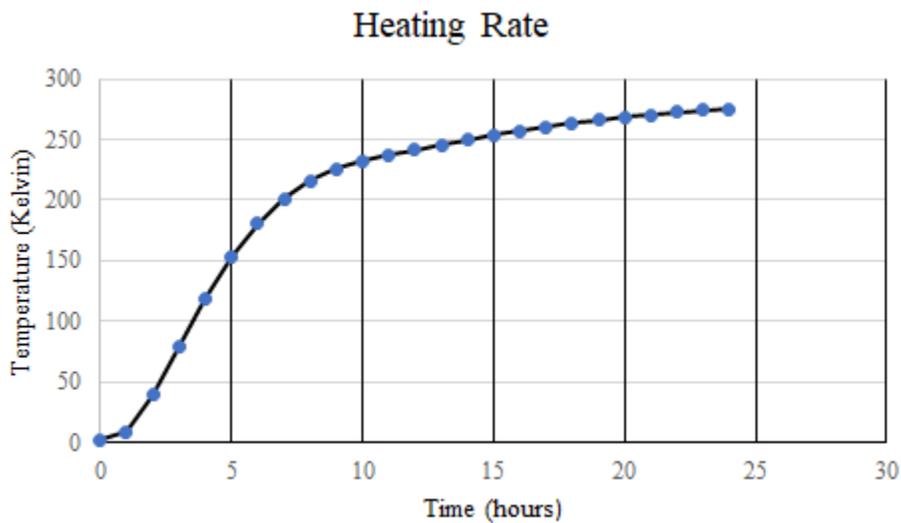


Figure 79: The rate of heating for the magnetorquer over 24 hours

The temperatures of the various components simulated in COMSOL after a period of 24 hours were recorded for both the hot and cold case and compared with the operating and survival

temperatures. The simulations of these internal components for both the hot and cold case are shown in Figure 75 and Figure 76 respectively. These values were viewed as indicative of the temperature of MagSTARS over the course of its mission as the temperature at this time is shown to level out as displayed in Figure 79 which demonstrates the temperature of the magnetorquer over the 24-hour simulation.

Table 37: Simulated Temperature Comparison

Component	Simulated Temp. Hot case[C°]	Simulated Temp. Cold case[C°]	Operating Temp. [C°]	Survival Temp. [C°]
Magnetorquers	2.11	-6.19	-40 to 85	-40 to 85
Magnetometer	3.81	-4.71	-40 to 85	-40 to 85
GPS	5.95	-1.52	-40 to 85	-40 to 85
Sun Sensor	3.81	-4.71	-25 to 70	-40 to 85
Control Board	-26.55	-36.19	-25 to 65	-40 to 85
Accelerometer	-29.24	-38.54	-40 to 93	-40 to 85
Angular Velocity Sensor	3.81	-4.71	-35 to 60	-40 to 85
Transceiver	-21.61	-30.01	-20 to 50	-40 to 85
Antenna	-61.37	-68.5	-10 to 45	-40 to 85
Science Payload	3.52	-2.67	-10 to 50	-40 to 85
Battery	-53.47	-65.11	-20 to 70	-40 to 85

Within the model it was shown that most of the components were well within their thermal limits however two of the more sensitive components, the transceiver, the computer board, and the antenna were shown to be below their thermal limit in both the hot and cold cases. The antenna

was shown to reach particularly low temperatures as it was placed on the side of MagSTARS opposite the Sun facing side. This is shown in Figure 77 and Figure 78. The battery was also shown to have a temperature well below the survival temperature, however, the battery used on board MagSTARS was a model with a built-in heater, and so additional thermal control was not needed.

In order to remedy this a thermal control method had to be chosen, with the final decision being a patch heater. This method was chosen as it would add no additional weight to MagSTARS as that would change the analyses undergone by the mechanical and propulsion subsystem. Another reasoning was that although it was an active method it would use little power in comparison to other methods of adding heat. The Wire Kinetics Ultra Heating fabric was the chosen patch heater [104]. This device was a 10 x 5 cm heating pad which made the pad an appropriate size for all of the components in need of thermal control as the antennas footprint on the mechanical model was 7 x 7 cm and the dimensions of the transceiver were 10 x 5 cm. The computer stack containing the control board was and 10 x 10 cm allowing for the pad to fit well within those dimensions. The pad was optimized to a temperature of 40 degrees Celsius by setting the pad to an input of 5 V and a current of 0.74 A. Due to limitations within the power subsystem the heaters could not be run at the same time as propulsion maneuvers. This development was ideal however as this stipulation would mean that the antenna and transceiver could run optimally during the Science Phase. This would allow MagSTARS to transmit the data it collected in the F layer as planned as the majority of the transmitting and data collection of the mission will occur during the Science Phase.

10 Helmholtz Cage

This chapter discusses the Helmholtz cage developed for magnetic field testing. It covers the theory behind Helmholtz cages, the design and construction of the cage, and the results of cage testing.

10.1 Helmholtz Cage Overview

A Helmholtz cage consists of three pairs of orthogonal coils which are mounted in such a way to create desired magnetic fields near the center of the cage. The Helmholtz cage is named after Hermann von Helmholtz (1821-1894), who contributed to the field of electromagnetism [25]. When a pair of wire coils is placed at a proper distance apart and current is run through the coils, a magnetic field is produced. This magnetic field will be of near constant magnitude and direction inside a volume between the two coils and can be used to simulate magnetic field conditions experienced by spacecraft. In most cases, coils are square to increase the volume of useable magnetic field and to make construction and use easier. The CAD model in Figure 80 below shows the basic design of a Helmholtz cage:

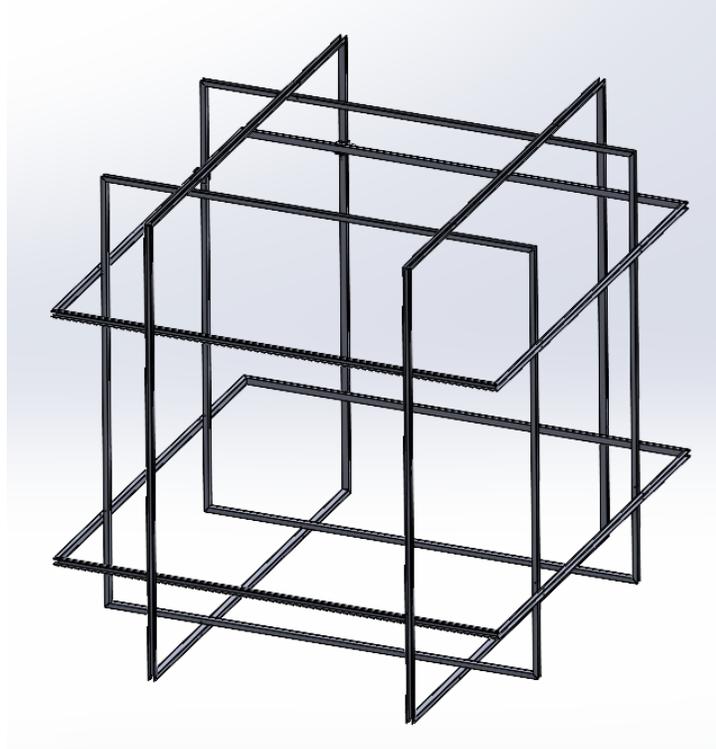


Figure 80: Basic Helmholtz cage design

10.2 Helmholtz Cage Development and Design

In order to determine the size, length of wire, and power requirements of the Helmholtz cage, targets for the magnetic flux density and constant field volume were set. The magnetic flux density on the surface of the Earth varies from approximately 0.25 to 0.65 Gauss [105]. As MagSTARS was in a low Earth orbit, the magnetic flux density experienced was similar to that present at the surface. Therefore, the Helmholtz cage had to be able to produce a minimum magnetic flux density of 0.65 Gauss in any direction. However, as the magnetic field of the Earth may have to be cancelled out by the cage, the maximum flux density produced had to be approximately double this value. The magnetic flux density in Worcester is approximately 0.52 Gauss (G), therefore, if a magnetic flux density of 0.65 G is desired in a direction opposite to the

local magnetic field, a total flux density of approximately 1.17 G is required [106]. From this analysis, a magnetic flux density target of 1.5 G was selected to provide a margin of error.

The longest side length of MagSTARS' 6U design was approximately 30 cm. In order to accommodate a full CubeSat of this size, and to allow for future testing with larger components, a maximum 1 percent field strength variation for 0.25 m in each direction along the centerline of each coil pair was selected. With three orthogonal coils this would produce a volume of near constant magnetic fields large enough to accommodate any test items similar in size to MagSTARS.

With these magnetic flux density and size goals set, an analysis of possible cage configurations was performed. To do this, an understanding of the equations governing the magnetic field of a Helmholtz cage was required. The Helmholtz cage theory is an application of the Biot-Savart law, which describes the magnetic flux density, \bar{B} , with the following equation:

$$\bar{B} = \frac{\mu_0 n I}{4\pi} \int \frac{d\bar{l} \times \hat{r}}{r^2} \quad (35)$$

Where μ_0 is the permeability of free space, $d\bar{l}$ is an infinitesimal wire element, and \hat{r} is the distance from the wire element to the point of magnetic field measurement. By applying this law to a pair of square coils, an equation for the magnetic flux density along the centerline of the coils can be written:

$$B_z = \frac{\mu_0 n I}{\pi} \left(\frac{2a^2}{\left(a^2 + \left(z - \frac{h}{2}\right)^2\right) \sqrt{2a^2 + \left(z - \frac{h}{2}\right)^2}} + \frac{2a^2}{\left(a^2 + \left(z + \frac{h}{2}\right)^2\right) \sqrt{2a^2 + \left(z + \frac{h}{2}\right)^2}} \right) \quad (36)$$

Where a is half the coil side length, z is the position along the central axis, measured from the central point between coils and h is the distance between the coils. At the central point of the coils, the equation for magnetic flux density can be simplified to the following form [26]:

$$B_z = \frac{2\mu_0 nI}{\pi a} \left(\frac{2}{(1+\gamma^2)\sqrt{2+\gamma^2}} \right) \quad (37)$$

Where γ is the ratio of the distance between coils and the coil side length.

Having established the equations that govern the magnetic field behavior of a square coil Helmholtz cage, two MATLAB scripts were developed to facilitate selection of cage specifications. Three design specifications were targeted through this analysis: cage dimensions, wire type and length, and power supply requirements. The power supply requirements were determined from the known current requirement to produce a desired magnetic field and an estimation of the wire resistance, R . The power supply voltage, V , and power, P , were easily determined using the following equations:

$$V = IR \quad (38)$$

$$P = IV \quad (39)$$

To find the minimum cage size necessary, the magnetic flux variation along the central axis of various cage side lengths was analyzed. Using the first MATLAB script, shown in Appendix L, it was determined that a minimum cage side length of approximately 1.44 m would be required to meet the flux strength variation target. Figure 81 shows the magnetic field behavior along the centerline of such a coil pair.

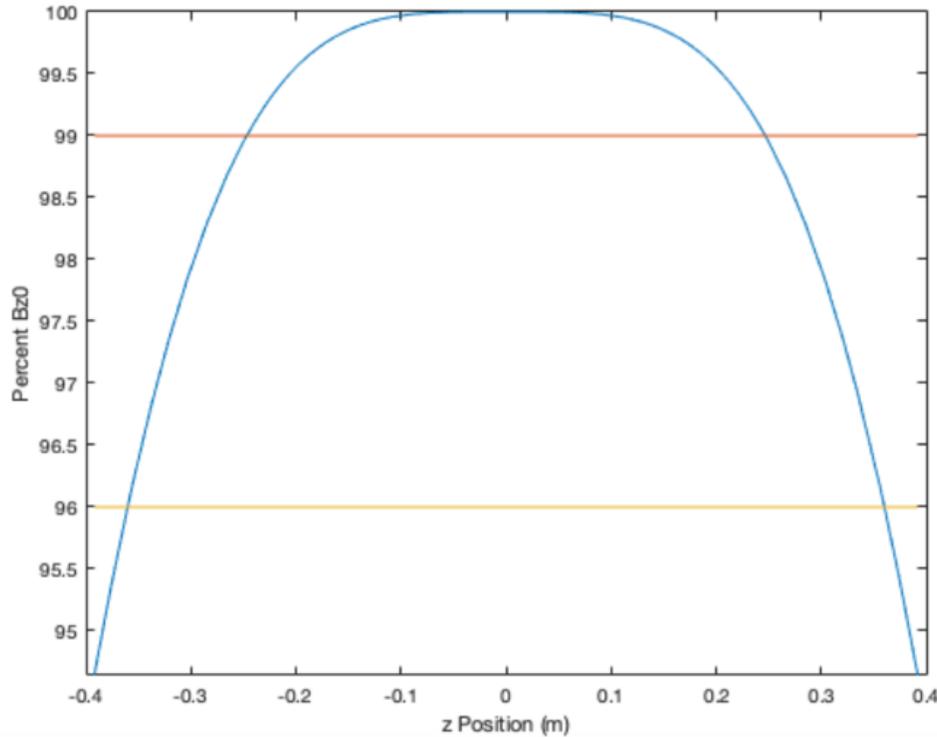


Figure 81: Magnetic field strength along the central axis of 1.44-meter coils

As shown in the figure, flux density within 1 percent of the central value is maintained for approximately 0.25 m in each direction. Reference lines at 96 and 99 percent flux density are included.

With a lower bound on the cage size established, the second MATLAB script, shown in Appendix M, was employed to iteratively analyze different configurations of coil size, current, wire type, and number of wire turns. Additionally, minimum or maximum acceptable values of current, voltage, power, wire length, and magnetic flux density were set. Coil sizes from 1.44 to 2 m were considered, with the upper bound derived from space constraints. Current values were limited to 4.75 A to accommodate the capabilities of the selected power supplies, discussed in detail in Section 10.3.2. Up to 60 turns of wire per coil were considered. Turn counts above this

number were deemed undesirable due to extra winding time during construction, as well as excessive cost due to greater wire length.

The type of wire used in Helmholtz cages is known as magnet wire and is typically made from solid copper wire covered with a thin layer of dielectric insulation. Different wire diameters characterized by the American wire gauge (AWG) standard were compared for use in the Helmholtz cage. In the AWG system, higher gauge numbers correspond to wires with smaller diameters. Drawing from knowledge gained during a literature review of Helmholtz cage designs, wire gauges from AWG 12 to AWG 22 were considered for use. A higher gauge wire uses less material per unit length and is therefore less expensive. However, it has a higher resistivity and requires more power to achieve a target current. Higher gauge wire also reaches higher temperatures, lowering the maximum current that can be supplied without melting the insulating material. Magnet wire is sold in various lengths, and as it was not desirable to splice two wires together, maximum wire length limitations were based on available spool size purchase options for various gauges. Additionally, it was not considered advantageous to use less wire if it did not enable a step down to a smaller spool size.

From the calculated maximum current of 4.75 A per coil, AWG 17 wire was selected. This wire gauge was estimated to withstand the thermal load induced by the current, while also being cheaper and available in greater lengths per spool than larger gauge wire. The temperature under load, and thus current carrying capacity, of a wire is dependent on the placement of the wire and local environmental conditions. It was therefore difficult to estimate the current limit without computational simulation. In order to select wire, the wire gauge and current used by the Air Force Institute of Technology and California Polytechnic teams were referenced. The AFIT team utilized AWG 12 wire at a maximum current of 7.52 A [25]. The California Polytechnic team initially

selected AWG 13 wire before revisiting their choice and moving to AWG 20 wire. They estimated this wire to have a current limit of 7.5 A, though a 50 percent safety factor was used for coil current of 3.75 A [107]. From this research, the team selected Remington Industries AWG 17 magnet wire.

Remington AWG 17 wire is sold with a maximum coil length of 1594 ft. The initial analyses showed that certain current and wire turn combinations would be able to produce the desired magnetic field conditions using one Remington AWG 17 coil per cage coil pair. A wire length limit of 1575 ft per coil pair was imposed in the MATLAB script, with the extra 19 ft left as safety factor. Following simulation with these conditions in place, it was determined that a maximum coil size of 1.75 m would be achievable. This size was selected to be used by the outer coil pair, with the two inner pairs being two smaller sizes. Figure 82 shows that only one coil configuration with a side length of 1.75 m was able to meet the magnetic field requirements.

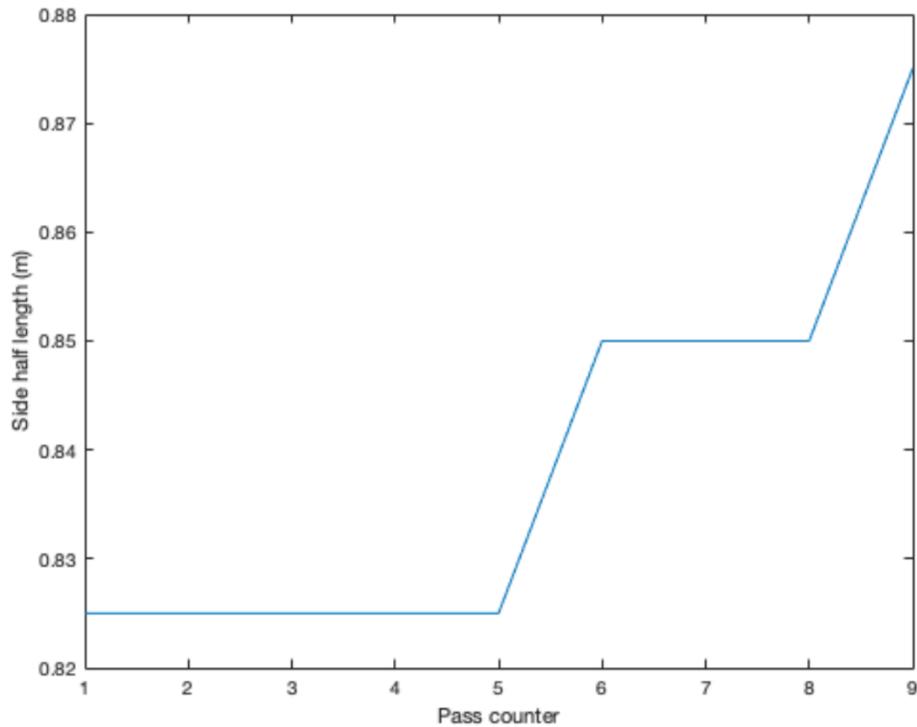


Figure 82: Number of cases that met the magnetic field requirements for a coil side length range of 0.825 to 0.875 m with a 0.025 m step size

10.3 Structural Design

With the coil dimensions determined, a structure was designed to support these coils. This structure was primarily constructed out of U-channel aluminum, chosen for its strength, ease of use, and non-magnetic properties.

When selecting the primary method for connecting U-Channel to form these coils, three primary options were considered. Using fasteners such as threaded bolts allowed for the coil itself to be disassembled after completion. However, due to the man-hours required to wind the magnet wire every time the coil is disassembled; this was not considered a major benefit. In addition, due to the nature of winding wire through the U-channel, bolting each corner would obstruct the

winding process and increase the difficulty of assembly. Holes would also need to be drilled for every fastener which would not be a trivial task for such a large structure. The coils also need to slide together to fully assemble the Helmholtz cage, and having bolt heads and mounting plates protruding from the outside edges of the U-channel causes assembly to be significantly more difficult, making bolts an impractical option for this application. Rivets were also considered as a possible mounting solution, as they are significantly lower profile than alternative threaded fasteners and are extremely easy to use. However, even with their reduced profile, rivets still obstruct wire placement and frame assembly. Welding was chosen as the primary attachment method, as it provides strong connections between U-channel segments while maintaining nearly zero protrusion beyond the profile of the U-channel both externally and within the channel. While this assembly method is more involved in terms of man hours required, the benefits to design functionality are worth the increased work. Wire winding is completely unobstructed with this method, drilling holes for fasteners is unnecessary, the coils external dimensions remain unchanged, and the resulting structure is extremely durable. U-channel was brazed together into three sizes of square coil, representing inner, middle, and outer nesting coils.

Several methods of supporting the corners of these coils were considered. The first method involved a triangle bracket bolted to each side of the corner. This method had the advantage of strength and ease of construction, given that the brackets could be constructed out of flat aluminum stock. However, it would require drilling holes through the main channel for attachment, possibly interfering with the magnet wire. For this reason, this method was not chosen. The second method discussed was using additional channel cut at 45-degree angles and brazed onto the structure. A possible addition to this method was a 3D-printed corner to better allow the wire to bend. This method did not require any additional metal parts, as it only used the same U-channel stock used

in the rest of the Helmholtz cage. It also did not interfere with the magnet wire or make the coil thicker. For these reasons, this method was chosen. A comparison of the corner attachment methods is shown below in Figure 83.

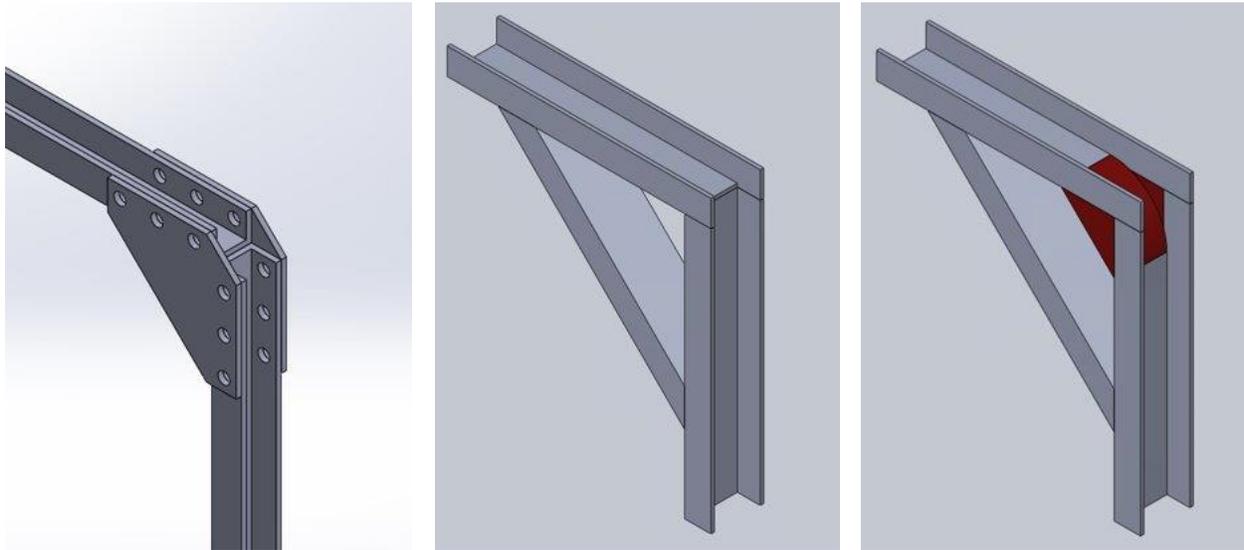


Figure 83: Helmholtz cage corner attachment methods

With the corner brackets determined, it was necessary to determine how the coils would be positioned in the correct positions. Several designs were again discussed. First, a cubic frame surrounding the entire Helmholtz cage was considered, similar to the Air Force Helmholtz cage [25]. Ultimately, this idea was discarded due to build complexity. As the channel was strong enough to stand freely, a superstructure was deemed to not be required. Then, it was necessary to determine how the coils would attach to each other and the ground. First, a rotating connector that would allow the Helmholtz cage to be folded up was considered, similar to the California Polytechnic Helmholtz cage [107]. This bracket would allow for easier storage but was very complicated to build.

A second option discussed was using a “shelf” attached to the vertical coils to support the horizontal coils. The vertical coils would be supported by the base. This option would provide simpler installation and removal of the coils, as they would only be directly attached to each other in limited ways. Problems arose when considering how to install the lower of the two horizontal coils, and if the structure would be stiff enough.

A third option discussed was using short pieces of channel that could tightly surround the main channel. Two pieces of channel would be brazed together, back-to-back, at 90-degree angles relative to each other. This would then allow two coils at 90 degrees to each other to be attached. The brackets would require a small cotter pin in order to stay in place, requiring a small hole to be drilled into the main channel. Ultimately, this method was chosen, as it was simpler to construct than the rotating brackets. A small rubber spacer was also added between the bracket and channel, which allowed easier disassembly. This design is shown below in Figure 84.

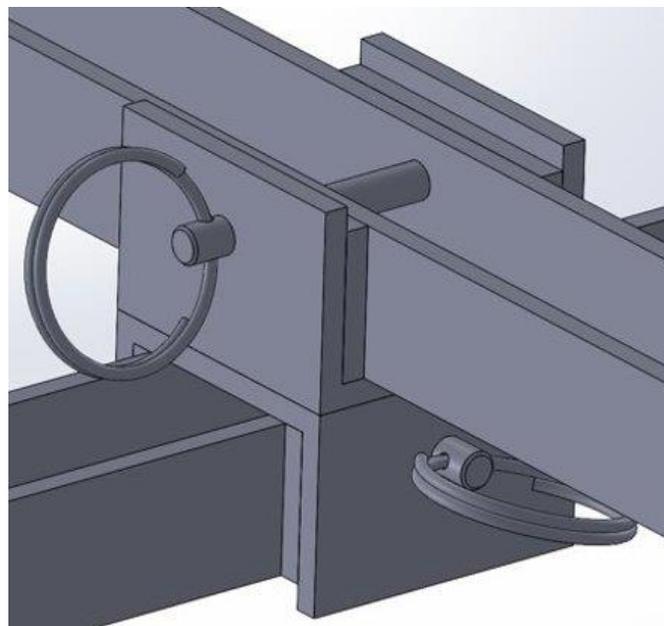


Figure 84: Helmholtz cage bracket

To disassemble the Helmholtz cage, one of the cotter pins in each of the brackets connecting the outer and middle coils should be removed. The outer coils can then be slid off the middle coils. The process is then repeated, with the inner and middle coils. This allowed the coils to be stored relatively flat, saving space.

One other structure designed for the Helmholtz cage was the base. Initially, a more complex base constructed of aluminum channel was considered, but when the bracket method was chosen, this was no longer necessary to fix the coils relative to each other. Thus, the final base design was simply constructed of wood.

10.3.1 Magnetorquer Mounting Method

In the low-friction, low gravity environment of space, applying a small amount of torque to an object can produce a significant amount of rotation. Because of this, MagSTARS' magnetorquer is designed to produce a small torque. However, on the surface of the Earth, where gravity is very much present, this presented a challenge. The low torque of the magnetorquer means resulting movement was difficult to detect, and the magnetorquer had to be mounted in a way that minimizes friction.

There were low friction mounts available, but many of these were complex and far outside the budget of this project. For example, industrial-grade air bearing tables can cost thousands of dollars and weigh hundreds of pounds. However, several low-cost options were considered, and it was ultimately decided to make an air-bearing table out of a portable air hockey table. The air hockey table is essentially a box with a fan on the underside, and the top is a slick material with many small holes to create a cushion of air and provide easy movement for the puck. A circle of lightweight plastic was used as the magnetorquer's stand and was lightly pinned down to the center

of the air hockey table to allow for rotation but no linear movement. If the air pressure is not high enough to provide the low friction surface the magnetorquer requires, the hockey table may be taken apart and retrofitted with a stronger fan. It was decided that the best way to test the effectiveness of this option was to buy the table and physically test it, as most air hockey games cost roughly \$30 and if necessary, pursuing another mounting option would not have a major impact on the budget.

10.3.2 Power Supplies

Power supplies were used to supply current to the wires in the cage in order to produce the magnetic field desired by using the commanded voltage from the connected computer and Arduino set-up. Kungber variable DC power supplies were selected for this project, shown below in Figure 85.



Figure 85: Kungber DC power supply [108]

These power supplies were capable of supplying 30 V at 10 A, which was appropriate based on the estimated coil current requirements and resistances [108]. However, these power supplies could not be controlled via the Arduino directly, necessitating the use of motor controllers.

10.3.3 Motor Controllers

The motor controllers were required to adjust the strength of the magnetic field created by the Helmholtz without the use of a variable power source. Motor controllers are an electrical component that is normally used to adjust the speed and direction of a motor, but within the design of the Helmholtz cage, these devices were used differently. In this design the motor controllers allowed for the manipulation of the current the Helmholtz cage received in order to ensure the Helmholtz cage could create adjustable magnetic fields. The motor controller was varied in speed in order to vary the voltage. This variance in voltage can then be used to manipulate the current in both direction and magnitude, which in turn would manipulate the magnetic field the Helmholtz cage generated. Three of these motor controllers were needed to account for each axis of the Helmholtz cage.

The controller chosen was the Cytron 30A, 5-30V Single Brushed DC Motor Controller, shown below, as this device met the requirements for current and voltage required by the Helmholtz cage design. The voltage input range of the device varied from 5 to 30 V. The motor controller operates at a current of 30 A continuously and had an 80 A peak [109]. The controller also offered overcurrent limiting and temperature protection which acted as an advantage over the Helmholtz cage designed by California Polytechnic Institute which cited overheating and burning wires due to the amount of current running through their motor controllers [107].



Figure 86: Cytron 30A, 5-30V Single Brushed DC Motor Controller

10.4 Magnetorquer Test Article

In addition to testing the operation of the Helmholtz cage through magnetic field measurements, the team decided to develop a magnetorquer test article for use in the cage. While the primary goal was validating the cage performance, implementing a test article also provided experience that could inform future research groups on how to best utilize the cage for component testing. The choice of a magnetorquer test article enabled the team to practice collecting various data types such as torque, angular acceleration, angular velocity, and magnetic field interaction.

For the test article, the team decided to produce a magnetorquer with high magnetic dipole strength using components the team already possessed. This included a coil of magnet wire, an Arduino Nano microcontroller, an 11.1 V LiPo hobby battery, and a L298N motor controller/H-bridge. The supporting electronics were set up to record the gyroscope data as the magnetorquer is powered to track its effect on the article's angular acceleration. Subsequently, this can be used to determine the torque generated by the prototype magnetorquer. The L298N is used to adjust coil power via pulse with modulation, as well as polarity. To achieve a desirable magnetic dipole

strength, the team began by analyzing the equation given in (40). The multiple of the number of coils turns n and coil vector area can be written in terms of the wire length L and coil radius r :

$$nA = \frac{L}{2\pi r} \pi r^2 \quad (40)$$

$$nA = \frac{Lr}{2} \quad (41)$$

Using this substitution, it is shown that the magnetic dipole strength, and thus possible torque, increases with an increase in the coil radius. As the wire length available was fixed, it was determined that the magnetorquer test article should be designed as an air core magnetorquer. This design would enable a greater coil radius than a torque rod magnetorquer.

10.5 Data Collection Methods

Data on the magnetic field produced by the Helmholtz cage and the movement of the magnetorquer will be collected using two magnetometers and an inertial measurement unit (IMU). One magnetometer will be mounted on a stationary section of the magnetorquer mount, as close to the magnetorquer as possible without being attached to it. This will give a stationary reading of the magnetic field produced by the Helmholtz cage, and will primarily be used for initial testing of the cage and to provide control data for the magnetorquer readings. The second magnetometer will be mounted on the magnetorquer itself and will provide data on the magnetic field that affects the magnetorquer.

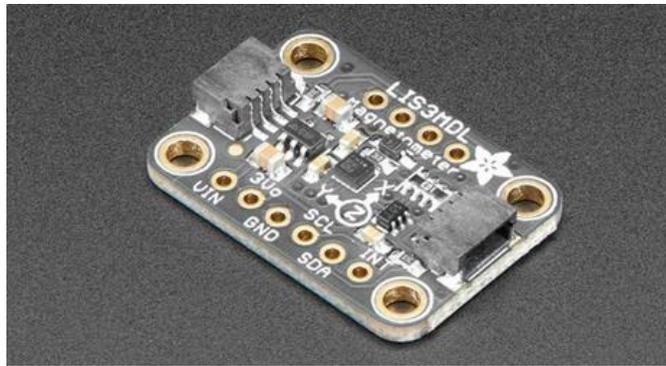


Figure 87: The Adafruit LIS3MDL Triple-Axis Magnetometer [110]

Adafruit's LIS3MDL Triple-axis Magnetometer, pictured in Figure 87, was chosen as the best option for this project. This will be used as the stationary magnetometer. It can measure magnetic fields from ± 4 G to ± 16 G, measures data at rates from 155 Hz to 1000 Hz, and costs \$ 5.95. It is not wireless, so wires will have to be accounted for in the mounting design. Bluetooth magnetometers are much more costly and more difficult to configure, so it was decided that a wired magnetometer was preferable [110].

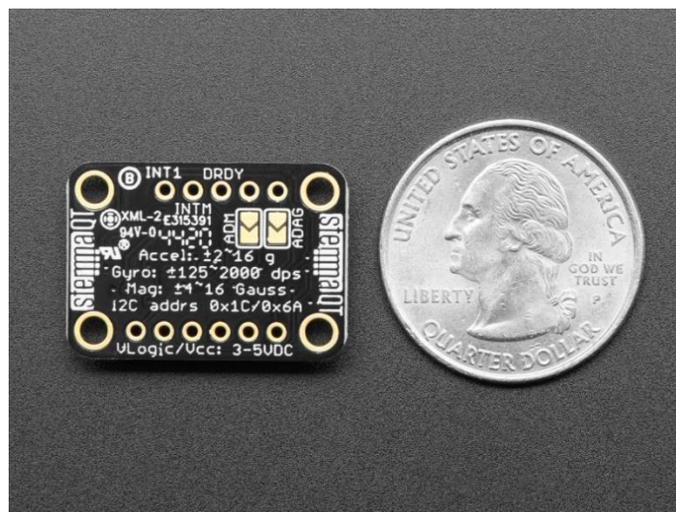


Figure 88: The Adafruit LSM6DSOX + LIS3MDL [111]

Adafruit sells a sensor that is a combined 6DOF IMU and 3-axis magnetometer, which was chosen to be mounted on the magnetorquer. The Adafruit LSM6DSOX + LIS3MDL, pictured in Figure 88, includes a LSM6DSOX 6DOF IMU accelerometer and gyroscope and measures linear and angular acceleration in 3D space. It also includes a LIS3MDL 3 – axis magnetometer, which is the same type of sensor that will be used to take control measurements. This combination IMU and magnetometer costs \$14.95. It must be hard wired, but the combination of the two sensors on one board allows for one less set of wires. [111].

Adafruit’s INA260 sensor was used to measure current in the magnetic coils. This sensor measures current, voltage, and power use on the high or low side, and is capable of measuring up to 36 V and 15 A, sufficient for this project [112]. Three of these sensors, pictured in Figure 89, were purchased: one for each power supply.

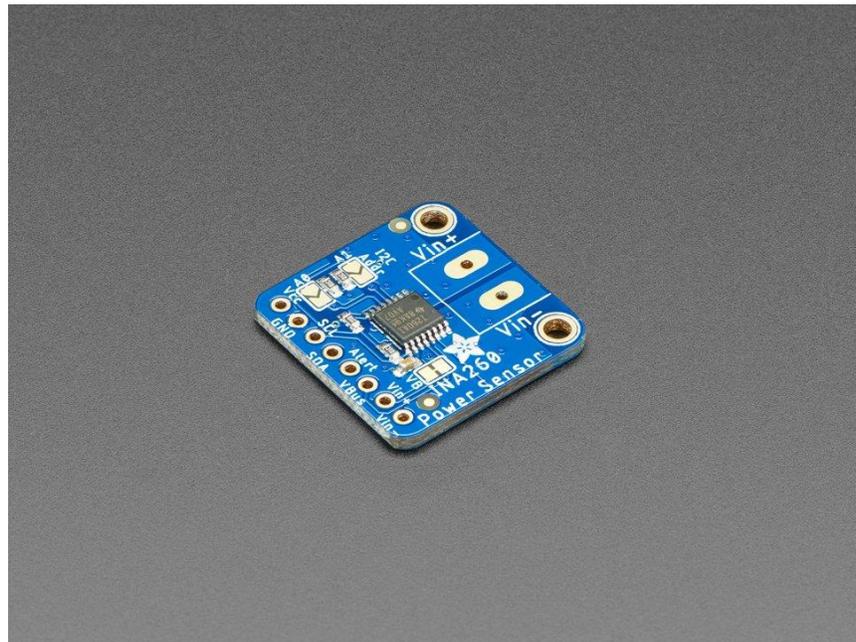


Figure 89: Adafruit INA260 Sensor [112]

One of these sensors was attached in series between each motor controller and the banana splitter leading to the coils.

10.6 Data Management

An Arduino MEGA 2560, shown below in Figure 90, was used to power and collect data from each sensor.



Figure 90: Arduino MEGA 2560 [113]

The Arduino MEGA supplied power at 3.3 V and 5 V, either of which was usable by the current sensors, magnetometers, and inertial measurement units. It also enables communication over the I2C protocol, used by each sensor. Each sensor has two inputs in addition to power and ground: SDA (I2C data channel), and SCL (I2C clock channel). By wiring each to the SDA and SCL pins on the Arduino, a number of sensors could be read simultaneously. The Adafruit sensors also had integrated pull up resistors, meaning no additional wiring had to be done beyond connecting each sensor input to the corresponding Arduino pin. However, as multiple of the same sensors were being used, additional steps had to be taken to ensure each device had a unique I2C address. The Adafruit sensors came with pads which, when bridged with solder, would change the

I2C address. Alternately, the corresponding pin could be wired directly to the Vin pin. A description of this is below in Table 38:

Table 38: Sensor I2C Addresses

Sensor	Pads Bridged	I2C Address
Current Sensor 1	None	0x40
Current Sensor 2	A0	0x41
Current Sensor 3	A1	0x44
Magnetometer 1 (IMU integrated)	ADM	0x1E
Magnetometer 2	None	0x1C

With each sensor now having a unique address, Arduino code was written to read in each sensor and output to the serial port. This code was based on the demonstration examples for each sensor and relied on the provided Adafruit libraries. It is attached in Appendix N. When completed, the readings of each sensor could be monitored through the Arduino serial monitor and plotter. However, this plotter was unwieldy, and it was also unable to write data to a file for future use. Thus, additional Python code was written to read the serial port using the PySerial library, write the data to a CSV file for later use, and plot the measured values in real time. This code is also attached in Appendix N.

In order to control the magnetic field strength of the cage, magnetic field data was stored in floating-point numerical arrays in the Arduino code. This data was then calibrated to correspond to a current strength and converted into PWM output for the attached motor controllers, allowing

modulation of the field even with fixed power supplies. A similar method was used to change the field strength of the magnetorquer test article. Documentation of this code is also provided in Appendix N.

10.7 Assembly

The first step in the assembly process was drilling and welding of the coil frame components and mounting brackets. Due to the size of the completed coil frames, it was necessary to drill the mounting holes in frame members before welding. All the mounting pin holes were drilled using a 7/32 in bit, allowing room for the 3/16 in cotter pins used to fix frame mounting brackets. Once all mounting holes were drilled, the coil frame members were prepared for welding by removing the anodized coating from regions to be welded using an abrasive disk tool, in order to facilitate high quality welds. The frames were subsequently welded together, as shown in Figure 91 along with the mounting brackets.

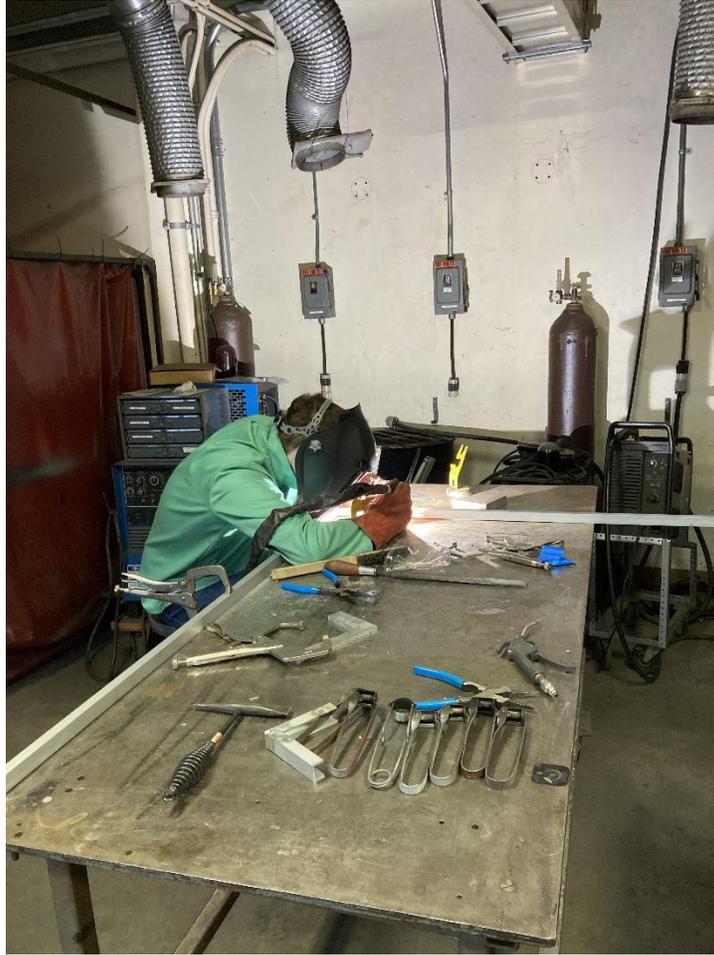


Figure 91: Coil welding in progress

To aid coil stability while welding and to improve the accuracy of the coil angles, the corner support pieces were welded onto the coil members as each coil corner was welded. A finished coil corner with support piece is shown in Figure 92.



Figure 92: Coil corner following welding

The coil frames were then readied for magnet wire winding by applying Kapton insulating tape along the entire wire channel as an extra preventative measure against the magnet wire shorting against the aluminum channel.

Once the inside of the coils were lined with Kapton tape, 17-gauge copper magnet wire was wrapped around each coil a specified number of times while each coil laid flat on a table. As explained in Section 10.2, the two largest coils were wrapped with wire 34 times, the middle-sized coils were wrapped 33 times, and the smallest coils were wrapped 32 times. With a spool of wire set in the middle of the table, the end of the wire was secured, and the wrapping began. The wire would be pulled off the spool as it was wound around the coil. The wire was pulled tight around each corner, as shown in Figure 93, ensuring it lie as flat against the coil as possible and trying to not overlap wires until surface area required overlap. Every 10 to 12 loops of wrapping, Kapton

tape was applied in two spots on each side of the coil to hold the wire down and prevent sagging. After the coil was wrapped with its respective number of wraps, the end of the wire from the spool was cut and secured.

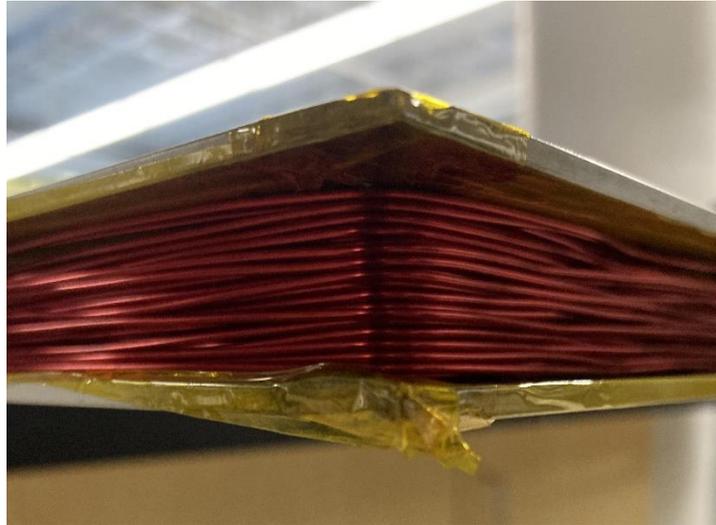


Figure 93: Wire at a coil corner

During assembly of the coils, the mounting brackets were also cut and welded together. Each bracket consisted of two small pieces of aluminum c-channel welded together at the base with the sides perpendicular to each other, as well as holes drilled through each side for the cotter pins to fit through. Rubber spacers were then superglued to the interior of one side of each piece of C-channel, as seen in Figure 94. Because the frames could not easily slide through the brackets with rubber spacers on each side, the other spacers had to be manually inserted. In order to easily keep track of these spacers, they were tied to the brackets with fishing line.

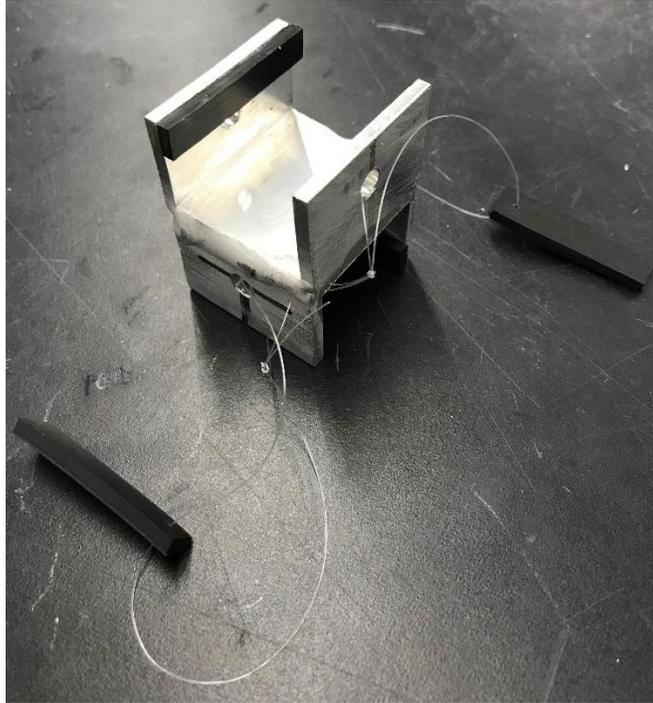


Figure 94: Mounting bracket with rubber spacers

Two small holes were drilled next to each other in each coil and the two ends of each wire coil fed through them. Kapton tape was used to protect the wire passing through the hole and prevent shorting to the frame. Female banana plug connectors were soldered to the ends of the wire as shown in Figure 95.

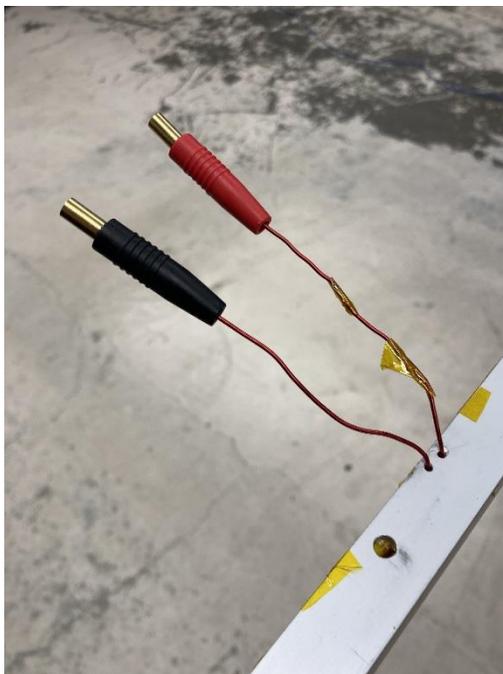


Figure 95: Coil banana plug connectors

Following this, the Helmholtz cage was ready for a full assembly. Following test fitting, minor adjustments were made to improve the alignment of some components. An image of the follow assembled cage structure is shown in in Figure 96.



Figure 96: Assembled Helmholtz cage

After coil winding was completed, 2 small holes slightly larger than the diameter of the magnet wire were drilled in the coil frame, and the ends of the coil wires threaded through. Subsequently, the enamel was removed from the tips of each coil wire, and female banana connectors were soldered on. Additionally, a thin layer of Kapton tape was wrapped around the magnet wire where it passed through the holes in the aluminum frame to ensure the wire would not short on the frame. 12 approximately 10 ft connecting cables were created using 12 AWG wire by soldering male banana plugs and covers to each end for connecting the coils to the power supply system. Using wire junctions, each coil set was plugged into its corresponding motor controller, with the 4 connecting cables from one coil set plugging into the 4 female plugs of the junction in the correct configuration, and the 2 male ends of the junction plugged into the motor controller

and current sensor via short connector wires soldered to female banana connectors. The motor controllers are connected to the power supplies via 6” positive and negative connector wires with one bare end for use in the motor controller terminal blocks, and a male banana connector wire soldered on for connection to the power supply.

10.8 Testing

To ensure the coil temperature would not reach levels that could damage the magnet wire during use, a maximum thermal load test was performed. A coil was powered, and 5 A run through it for a period of 20 minutes. A temperature reading was taken every minute during this period using a multimeter thermocouple, as shown in Figure 97.



Figure 97: Helmholtz cage temperature reading

Over the 20-minute period of measurements, the temperature reached a maximum of 120 °F. Though the temperature was still rising at approximately 1 °F per two minutes, the true maximum temperature would not be significantly higher than 120 °F. This temperature is well below the wire temperature rating of 311 °F [114]. A plot of the temperature over time during the test is shown in Figure 98.

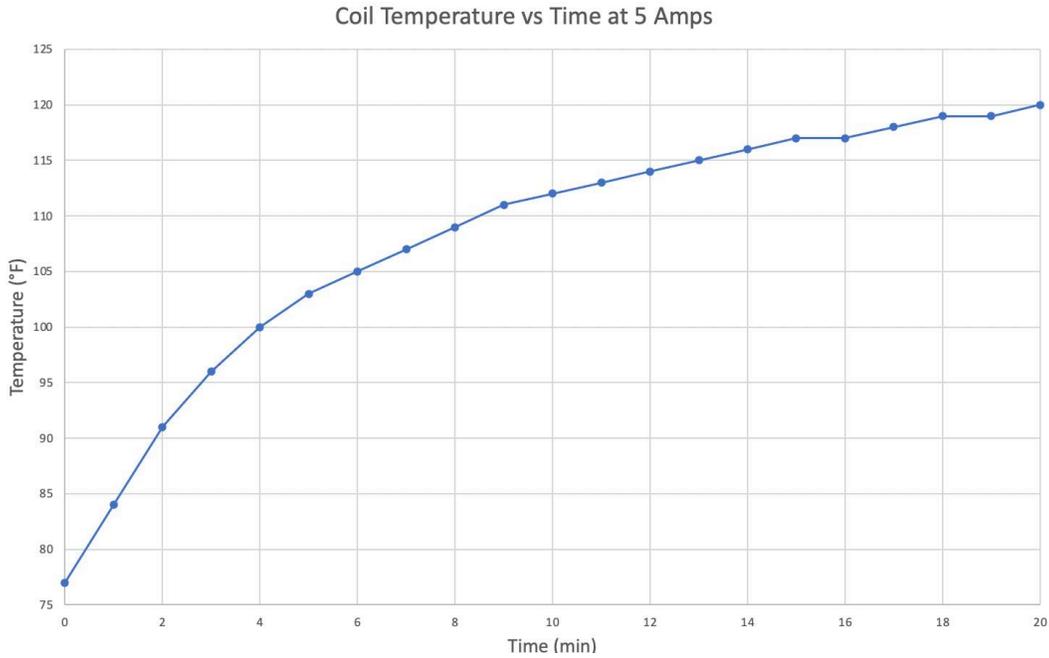


Figure 98: Helmholtz cage temperature plot

To evaluate the performance of the Helmholtz cage, test data was generated using STK. A scenario with sun-synchronous orbit similar to MagSTARS' orbit was produced and a report of the magnetic field in the body axes of the satellite was generated. A step size of one second was used for the output magnetic field data for one hour of the orbit. This step size would provide good resolution while not exceeding the limited memory of the Arduino. Depending on the test article being used in the Helmholtz cage, the axes for the output data would vary. If a fixed article were used, the body axes data should be employed while the Earth Centered Inertial (ECI) data should

be used for a free moving article. For the initial testing performed, the coordinate system used was not relevant. The STKL report was exported as a .csv file from which the magnetic field data would be copied and pasted into the Arduino code.

With the magnetic field data entered, the code was uploaded to the Arduino and the Python code was run to view live plots and produce a .txt file containing the recorded data. The program was run for approximately 50 minutes of the 60 total minutes of data. To ensure the magnetic field generation and measurements were not affected, the cage structure was not touched during the testing process. Following the completion of the test, the collected magnetic field data were plotted against the desired field data as shown in Figure 99:

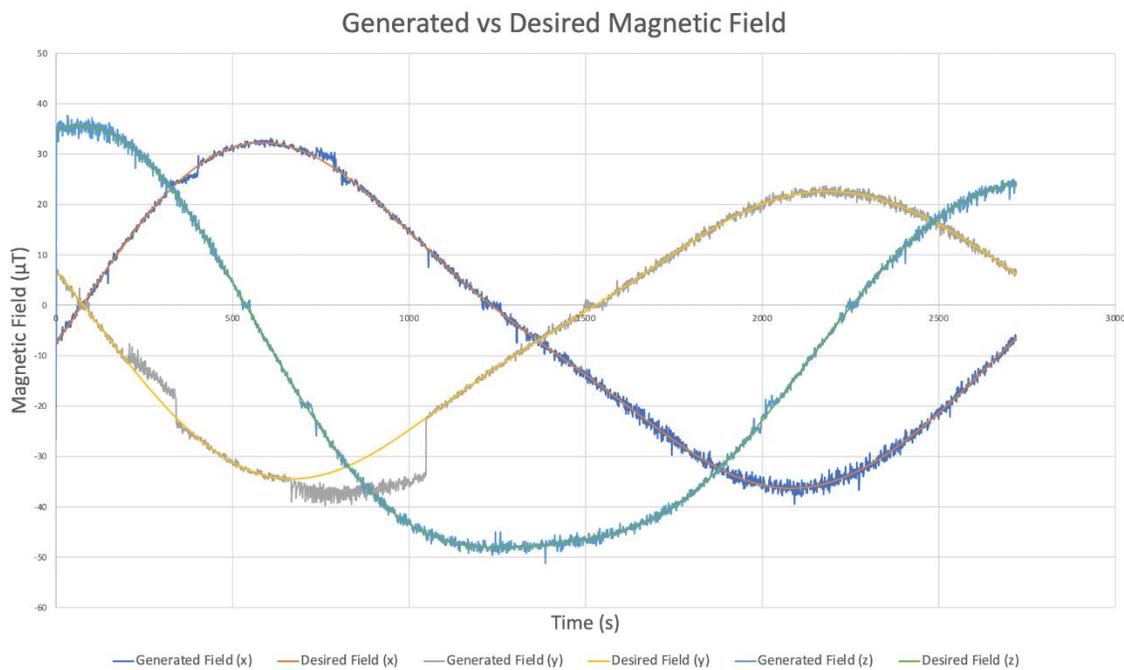


Figure 99: Helmholtz cage magnetic field test

There were multiple pieces of information to take away from these results. Of primary interest were the time periods during which the generated field did not map properly to the desired field. This was noted primarily on the y axis data, although some less significant behavior was also

be noted on the x axis data. The cause of these errors was not fully understood but was believed to be fully induced by errors in the setup of the Arduino control scheme. Corrections to the code should remove these errors.

During periods of proper behavior, the generated magnetic field matched the desired magnetic field well. For the z axis, analyzed because of the lack of the previously discussed errors, the mean absolute variance from the desired magnetic field was found to be $0.546 \mu\text{T}$. Additionally, the total variance was only $0.010 \mu\text{T}$. This indicated that over an extended period the average magnetic field experienced by a test article would very closely approximate the desired magnetic field conditions. The noise that caused the magnetic field variations seen in the figure had multiple sources. The local magnetic field in any location constantly varies. Additionally, minor variations in the current being supplied to the coils would affect the magnetic field. Lastly, magnetometer measurement error not only produced error itself but also affected the desired current output and thus the generated magnetic field.

The stand for the magnetorquer test article was constructed using a tabletop air hockey table. A piece of cardboard was loosely pinned to the middle of the table to allow it to rotate while staying in the center of the table. The extra holes in the table were covered with duct tape to direct all the air to the holes under the cardboard.



Figure 100: Magnetorquer test article stand constructed from an air hockey table

Unfortunately, the air pressure from the fan was not enough to lift the cardboard off the surface, much less lift the entire test article. Increasing power to the fan or minimizing the interior volume of the table could provide more air pressure.

11 Conclusion, Recommendations, and Social Implications

The work completed during this project has provided the team with a greater insight into the process of building a real spacecraft, as well as several recommendations for similar projects at WPI moving forward.

11.1 Conclusions

The MagSTARS mission is designed for an elliptical orbit in the F layer of the ionosphere, collecting particle composition data using the mini-INMS payload in order to better understand the makeup and variations of the F layer. MagSTARS is conceptualized as a spinner spacecraft with a magnetorquer as a primary means of attitude control and is designed to optimize mission life. The project considered structural design, propulsion selection, orbital analysis, power system design, telecommunications analysis, attitude determination and control system design, environmental effects analysis, and thermal analysis.

MagSTARS is baselined to the Planetary Systems Corporation Canisterized Satellite Dispenser (CSD), which utilizes the tab-based method for containing CubeSats. This allows for more accurate analysis of vibrational characteristics, to ensure that the CubeSat will maintain structural integrity during the launch environment. MagSTARS conformed to a number of mechanical requirements, from the CubeSat specification, the CSD payload specification, the NASA General Environmental Verification Standard, and the SpaceX Rideshare User Manual.

MagSTARS uses a BET-1mN thruster with a tank three times its original tank size, providing 0.258 kg of propellant for orbital insertion, orbit maintenance, and margin of error. Using this propellant, MagSTARS will be able to maneuver from the 625 km deployment altitude to the Science Leg orbit band with an apogee altitude range of 350 km to 400 km and a perigee

altitude range of 200 km to 250 km, and can maintain this orbit for 120 days by performing three stationkeeping maneuvers.

The power system selected for MagSTARS is the ISISpace MEPS, and a series of two 1x3 U, one 2x6 U and one 1x2 U solar arrays, based on performance compared to other systems. Analysis determined the power system will operate nominally throughout the mission, within max battery DOD range and provide all power required by each subsystem for each mission phase.

A telecommunications system was designed for optimal transmission and receiving capabilities. MagSTARS will have approximately 7 minutes of access time per access, with multiple connections per orbit. Each access has more than enough time to transmit or receive the data from the science payload or the ground station.

To achieve its mission goals, an attitude determination and control system for MagSTARS was designed. Potential components for this system were analyzed through decision matrices and best components for the CubeSat selected. A B-dot detumbling controller was designed and implemented to reduce initial angular rates following deployment of MagSTARS as well as an attitude determination program using the TRIAD method. To achieve the scientific tumbling motion, work into developing an attitude controller was performed. Multiple approaches for this attitude controller were investigated to solve this complex problem. For future work on magnetorquer only control schemes, continued development of the trajectory optimization controller is recommended.

Analysis of the effects of the space environment on MagSTARS throughout its mission was conducted. This analysis included modeling the effects of magnetic field induced by the magnetorquers on MagSTARS' magnetometer, calculating MagSTARS' total ionizing radiation dose as well as the likelihood of damage from SEPs and GCRs, and calculating the potential

damage from particle impacts. Overall, MagSTARS' short mission life and small size meant no extra shielding or mitigation strategies against any of these factors was required, aside from a backup magnetometer.

By employing patch heaters, the thermal control of MagSTARS will be able to maintain the operating and survival temperatures of sensitive components on board. The variance in temperature over the course of the mission due to changes in albedo and solar flux were investigated but were determined to result in relatively small changes in the final temperature of the CubeSat.

Finally, a Helmholtz cage needed for magnetic field testing was designed, and fully constructed during this project. This cage was housed in HL016.

11.2 Recommendations

Working on this project provided the team with recommendations for areas of improvement for future MQPs considering CubeSat development. This section outlines these recommendations.

11.2.1 Spacecraft Charging

As discussed in Section 8.1.3.2, spacecraft charging is an important environmental effect that must be considered for this mission. For this project, spacecraft charging analysis was superseded by analysis of other environmental factors. The team suggests that in-depth spacecraft charging analysis be added to this mission design in future projects now that other basic environmental effects analysis has been completed.

11.2.2 Integrated System

During the course of the MQP, several manufacturers of CubeSat components were investigated. While components were eventually chosen from many different suppliers or designed by the team in the case of the CubeSat structure, it became clear that nearly an entire CubeSat could be constructed using components from a single supplier such as ISISpace. The team recommends that future CubeSat MQPs strongly consider this option, as it allows for guaranteed interoperability, simplicity of construction, and more available documentation.

11.2.3 Magnetorquers and Spinner Satellites

This CubeSat used magnetorquers as the only ADCS actuator. While this has significant advantages, including no required fuel, it also has a significant disadvantage in that torque can only be produced orthogonal to the local magnetic field. This makes achieving specific pointing goals at arbitrary times and locations challenging. In order to have a spinning satellite, the satellite must be able to spin up and down as required for data collection and propulsion. The team recommends that future teams continue research into the trajectory optimization controller for magnetorquer only control. However, it is also recommended that additional methods of attitude control such as reaction wheels or thrusters be considered.

11.2.4 Thermal analysis

There were several aspects of the thermal analysis that could be improved upon in future research. The hot and cold case of MagSTARS proved to be of minimal importance in terms of determining whether or not the components on board would be within the range of their operating or survival temperature. Ultimately it was found that there was very little variance between these two cases. Future missions should find adequate simulation simply by using the average albedo of

Earth and the average solar constant in their calculations. Another possible improvement is extending the time in which this thermal analysis is performed. The MagSTARS flight plan was quite long but thermal analysis could only reasonably be taken for short periods due to extremely long computation times. Future MQPs should seek out more sophisticated software and modelling tools to allow for much longer portions of the mission to be simulated. This will ensure the spacecraft has adequate thermal shielding in the event of longer missions.

11.2.5 Magnetorquer Test Article Stand

Because of resource and time constraints, the team was unable to build a working stand for the magnetorquer test article to rotate freely on within the Helmholtz cage. The fan within the air hockey table originally used for this purpose did not generate enough air pressure to lift the cardboard off the surface of the table, much less the test article. The team recommends improving on the air hockey table design for the stand by increasing power to the fan or replacing the fan entirely. The design could also be improved by constructing a smaller “air hockey table” in which the only surface the fan is blowing against leads directly to the rotating cardboard, which would also increase the air pressure. A full stand should also be constructed to elevate the test article to the right height and location within the cage.

11.3 Social Implications of CubeSats

The relatively low cost and simple, customizable design of CubeSats enables a variety of educational, commercial, and scientific benefits that are not feasible with traditional larger satellites. Because of their accessibility, CubeSats allow groups outside of the traditional satellite development organizations to engage in space-based operations. Educational institutions from universities to K-12 schools can use the development of a CubeSat as an excellent hands-on

learning experience. Small businesses and start-ups can meet the significantly reduced budget requirements a CubeSat offers over a traditional larger satellite. The accessibility of CubeSats opens the door to the space industry for many people who would not otherwise have the chance to contribute to it [9].

When developed in an educational setting, CubeSats offer students opportunities to participate in problem-oriented group work, and to gain experience that can be applied in the industry. In 2003, students from Denmark's Aalborg University launched the AAU-CubeSat, an early demonstration of CubeSats' use of imaging technology. The secondary task of this mission was to give students hands-on experience with a multi-disciplinary project. The students found working on the AAU-CubeSat to be a "highly motivating and unique experience." The technical challenges and inter-group coordination required to design, build, test, and fly the satellite gave students relevant experience they felt would benefit them in the workforce, and many were exposed to the necessity of strong project management [115]. The students involved in a more recent educational CubeSat mission, Aalto University's Aalto-1 CubeSat, found similar benefits of the CubeSat development process. The multi-payload satellite involved over 80 students across multiple years. Students said the project exposed them to multidisciplinary group work, taught them how to document their work correctly and efficiently, and fostered a sense of community and entrepreneurship within the project teams. [116]

CubeSats also have many commercial benefits. Large aerospace companies, like Boeing and Aerojet RocketDyne, can develop and test new technologies on CubeSats without having to invest in a costlier large satellite. Once developed and tested, the smaller, low power components created for CubeSats can also be used on larger satellites, increasing their overall efficiency. [117] [118]. Nonprofits and startups can also take advantage of these low-cost missions, like The

Planetary Society's LightSail project. This crowdfunded CubeSat successfully tested a solar sail as a means of propulsion in 2019, and LightSail 2 is still in orbit today [119]. CubeSats also offer gateways to space science for emerging and developing nations. Participation in the space industry has been shown to be a worthwhile long-term investment for increasing intellectual capital and developing new technology, but for many nations who struggle with providing access to basic resources, the price tag of space is too high. CubeSats give these nations the opportunity to take the first steps toward building a space industry of their own [120].

The educational and commercial accessibility of CubeSats allows for a new class of scientific innovation. CubeSats are the perfect vessel for demonstrating new technology, like LightSail's solar sail or the Multi-Application Survivable Tether Experiment (MAST). Space tethers have been developed for years with varying degrees of success. MAST was able to test and record data on this high-risk technology without risking a more expensive satellite, gathering invaluable information that can be used to improve space tethers in the future [121]. Along with being a platform for new technologies, CubeSats' accessibility allows a more diverse pool of researchers to participate in space science. This influx of new groups, perspectives, and backgrounds has allowed the space industry to grow in new directions. CubeSats bring the excitement and opportunities of spaceflight "closer to home than ever" [9].

MagSTARS was able to be created because of the many opportunities CubeSats provide, particularly in the educational sector. As a project concept, MagSTARS was extremely beneficial to the team in providing a realistic design challenge that explored the work done in the space industry every day. Learning about the different requirements, considerations, and challenges of spacecraft design through project work has provided the team with valuable experience that can be applied to future careers.

Its science mission has broader impacts as well. MagSTARS will collect particle density data at different latitudes in order to learn more about the variations and anomalies of the ionosphere. This is important to understand because radio and GPS communication rely on the ionosphere to work, as described in Section 8.1. Unexpected anomalies can cause disruptions in this communication, so it is important to gather data on these phenomena in order to predict disruptions and work towards adapting communications technology accordingly [87].

Bibliography

- [1] CubeSat, "The CubeSat Program," CubeSat, [Online]. Available: <https://www.cubesat.org/about>. [Accessed 6 10 2020].
- [2] NASA, "Small Spacecraft Technology," Ames Research Center, Moffet Field, 2020.
- [3] NASA, "About CubeSat Launch Initiative," 28 May 2020. [Online]. Available: <https://www.nasa.gov/content/about-cubesat-launch-initiative> . [Accessed 6 October 2020].
- [4] E. Kulu, "Nanosats Database," 4 October 2020. [Online]. Available: <https://www.nanosats.eu/>. [Accessed 6 October 2020].
- [5] J. Klenzing, R. L. Davidson, G. D. Earle, A. J. Halford, S. L. Jones, C. R. Martinis, N. Paschalidis, R. F. Pfaff, J. M. Smith and K. A. Zawdie, "petitSat - A 6U CubeSat to examine the link between MSTIDS and ionospheric plasma density enhancements," in *AGU 100 Advanced Earth and Space Science Fall Meeting*, San Francisco, California, 2019.
- [6] T. Andreani, S. Joy, E. Beerbower, B. Snyder, R. Clavijo, J. Valero and G. Lee, "Design and Analysis of a 6U CubeSat and Mission," Worcester Polytechnic Institute, Worcester, Massachusetts, 2020.
- [7] R. Galliath, O. Hasson, A. Montero, C. Renfro and D. Resmini, "Design and Analysis of a CubeSat," Worcester Polytechnic Institute, Worcester, Massachusetts, 2020.
- [8] SpaceX, "SpaceX Satellite Rideshare," [Online]. Available: <https://rideshare.spacex.com>. [Accessed 26 September 2020].

- [9] J. Howard, "CubeSats: Shaping Possibilities in Space," NASA, 6 August 2017. [Online]. Available: https://www.nasa.gov/mission_pages/station/research/news/cubesats_possibilities. [Accessed 3 October 2020].
- [10] SpaceX, "Rideshare Payload User's Guide," June 2020. [Online]. Available: https://storage.googleapis.com/rideshare-static/Rideshare_Payload_Users_Guide.pdf. [Accessed 15 October 2020].
- [11] NASA, "CubeSat Launch Initiative Selections," NASA, 22 May 2020. [Online]. Available: <https://www.nasa.gov/content/cubesat-launch-initiative-selections>. [Accessed 3 October 2020].
- [12] M. Safyan, "NanoRacks-PlanetLabs-Dove," NASA, [Online]. Available: https://www.nasa.gov/mission_pages/station/research/experiments/explorer/Investigation.html?#id=1176. [Accessed 3 October 2020].
- [13] D. M. Velez, E. I. Dawson and N. E. Nassiff, "Attitude Determination and Control Subsystem Design for a CubeSat," Worcester Polytechnic Institute, Worcester, 2012.
- [14] C. H. Dopart, E. S. Oliver, J. J. Schomaker and R. C. Morlath, "Design and Analysis for a CubeSat Mission," Worcester Polytechnic Institute, Worcester, 2012.
- [15] M. A. Carter, "Mechanical, Power, and Thermal Subsystem Design for a CubeSat," Worcester Polytechnic Institute, Worcester, 2012.
- [16] D. R. Billings, F. S. Hoey, I. S. Graedel and J. Michael, "Design and Analysis for a CubeSat Mission," Worcester Polytechnic Institute, Worcester, 2013.

- [17] A. T. Snapp, A. T. Farhat, J. T. Ivase and Y. Lu, "Attitude Determination and Control System for CubeSat," Worcester Polytechnic Institute, Worcester, 2013.
- [18] B. J. Joseph, J. M. Hanley, J. R. Trudeau, M. O. Miller, R. L. Weinrick and e. al., "Thermal, Telecommunication and Power Systems for a CubeSat," Worcester Polytechnic Institute, Worcester, 2013.
- [19] E. A. Curci, J. B. Jacobson, K. M. Slabinski and W. T. Schlack, "Design and Analysis for the Sphinx-NG CubeSat," Worcester Polytechnic Institute, Worcester, 2017.
- [20] D. J. Pelgrift, D. H. Ko, M. F. Murphy, S. A. Young and S. C. Laudage, "Design and Analysis of the Sphinx-NG CubeSat," Worcester Polytechnic Institute, Worcester, 2017.
- [21] A. M. Rathbun, J. A. Agolli and J. Gadoury, "Design and Analysis of the Sphinx-NG CubeSat," Worcester Polytechnic Institute, Worcester, 2017.
- [22] A. B. Kant, J. J. Nichols, L. J. Mancinelli, M. J. Knodler, M. F. C. Abad, M. Escalante-Hurtado and R. C. Crockett, "Design and Analysis of CubeSats in Low Earth Orbit," Worcester Polytechnical Institute, Worcester, 2018.
- [23] A. M. Koubeck, C. Maki, M. T. Sanchy and T. Winship, "Design of CubeSats for Formation Flying & for Extreme Low Earth Orbit," Worcester Polytechnic Institute, Worcester, 2018.
- [24] C. E. Lopez, G. S. Jacobson, J. N. Peters, N. A. Bograd and P. J. Kroyak, "Design and Analysis for CubeSat Missions," Worcester Polytechnic Institute, Worcester, 2018.
- [25] M. R. Brewer, "CubeSat Attitude Determination and Helmholtz Cage Design," Air Force Institute of Technology, Greene County, 2012.

- [26] D. T. Gerhardt, "Small Satellite Passive Magnetic Attitude Control," University of Colorado, 2014.
- [27] R. Cardoso da Silva, U. Rodrigues, R. Borges, M. Sampaio, P. Beghelli, S. Gomes Paes da Costa, B. Popov, S. Battistini and C. Cappelletti, "A Test-Bed for Attitude and Determination Control of Spacecrafts," *II Latin American IAA CubeSat Workshop*, 2016.
- [28] California Polytechnic State University, "6U CubeSat Design Specification Revision 1.0," 2018. [Online]. Available: https://static1.squarespace.com/static/5418c831e4b0fa4ecac1bacd/t/5b75dfcd70a6adbee5908fd9/1534451664215/6U_CDS_2018-06-07_rev_1.0.pdf. [Accessed 15 October 2020].
- [29] Planetary Systems Corporation, "Canisterized Satellite Dispenser (CSD) Data Sheet," 3 August 2018. [Online]. Available: <https://www.planetarysystemscorp.com/wp-content/uploads/2014/07/2002337F-CSD-Data-Sheet.pdf>. [Accessed 15 October 2020].
- [30] Planetary Systems Corporation, "Payload Specification for 3U, 6U, and 12U," 2018. [Online]. Available: <https://www.planetarysystemscorp.com/wp-content/uploads/2018/08/2002367F-Payload-Spec-for-3U-6U-12U.pdf>. [Accessed 15 October 2020].
- [31] F. Azure, R. Hevner and W. Holemans, "Lessons learned measuring 3U and 6U payload rotation and velocity when dispensed in reduced gravity environment," Planetary Systems Corporation, Silver Spring, 2015.
- [32] Planetary Systems Corporation, "Flight Heritage," 13 July 2020. [Online]. Available: <https://www.planetarysystemscorp.com/flight-heritage/>. [Accessed 15 October 2020].

- [33] NASA Science, "Basics of Space Flight," [Online]. Available: <https://solarsystem.nasa.gov/basics/chapter5-1/>. [Accessed 15 October 2020].
- [34] Clyde Space, [Online]. Available: <https://www.aac-clyde.space/>. [Accessed 13 10 2020].
- [35] ISIS Space, 12 1 2020. [Online]. Available: <https://www.isispace.nl/>. [Accessed 13 10 2020].
- [36] Endurosat, 12 10 2020. [Online]. Available: <https://www.endurosat.com/?v=7516fd43adaa>. [Accessed 13 10 2020].
- [37] NASA, "CubeSat101 Basic Concepts and Processes for First-Time CubeSat Developers," NASA, 2017.
- [38] European Space Agency, "Satellite Frequency Bands," ESA, [Online]. Available: http://www.esa.int/Applications/Telecommunications_Integrated_Applications/Satellite_frequency_bands. [Accessed 13 10 2020].
- [39] K. LaBel, "Radiation Effects on Electronics 101: Simple Concepts and New Challenges," in *NEPP Webex*, 2004.
- [40] NOAA, "Satellite Drag," [Online]. Available: <https://www.swpc.noaa.gov/impacts/satellite-drag>. [Accessed 4 October 2020].
- [41] R. Ladbury, "Radiation Environment Effects on Spacecraft," in *NASA Ames Research Center Radiation Meeting*, Mountain View, California, 2017.
- [42] D. Hyun Ko, S. Laudage, M. Murphy, D. Pelgrift and S. Young, "Design and Analysis of the Sphinx-NG CubeSat," Worcester Polytechnic Institute, Worcester, 2017.

- [43] J. Klenzing, R. L. Davidson, S. L. Jones, C. Martinis, K. A. Zawdie, G. D. Earle, J. M. Smith, A. J. Halford, S. Noel, N. Paschalidis, R. F. Pfaff and E. Robertson, "The petitSat mission – Science goals and instrumentation," *Advance in Space Research*, vol. 66, no. 1, pp. 107-15, 2020.
- [44] M. Rodriguez, N. Paschalidis, S. Jones, E. Sittler, P. U. D. Chornay and T. Cameron, "Miniaturized Ion and Neutral Mass Spectrometer for CubeSat Atmospheric Measurements," NASA Goddard Space Flight Center , Greenbelt, Maryland, 2016.
- [45] NASA, "Upcoming ELaNa CubeSat Launches," 10 August 2020. [Online]. Available: <https://www.nasa.gov/content/upcoming-elana-cubesat-launches>. [Accessed 6 October 2020].
- [46] M. Hatfield, "Dellingr: The Little CubeSat That Could," NASA, 23 October 2018. [Online]. Available: <https://www.nasa.gov/feature/goddard/2018/dellingr-the-little-cubesat-that-could>. [Accessed 2 October 2020].
- [47] M. Rodriguez, N. Paschalidis, S. Jones, E. Sittler, D. Chornay, P. Uribe, T. Cameron and B. Nanan, "A Compact Ion and Neutral Mass Spectrometer for CubeSat/SmallSat Platforms," in *Small Satellite Conference*, 2015.
- [48] C. Anderson, R. Cuerdon, B. Kelsey and N. Petilli, "Design and Analysis for a CubeSat Mission-III," Worcester Polytechnic Institute, Worcester, Massachusetts, 2020.
- [49] COMSOL , "Multiphysics Cyclopedia," COMSOL, 21 January 2017. [Online]. Available: <https://www.comsol.com/multiphysics/mesh-refinement>. [Accessed 2 December 2020].

- [50] NASA, "General Environmental Verification Standard," 22 April 2013. [Online]. Available: <https://standards.nasa.gov/standard/gsfsc/gsfsc-std-7000>. [Accessed 11 March 2021].
- [51] SpaceX, "SpaceX Falcon User's Guide," April 2020. [Online]. Available: https://www.spacex.com/media/falcon_users_guide_042020.pdf. [Accessed 10 March 2021].
- [52] J. Blandino, N. Martinez-Baquero, N. Paschalidis, N. Gatsonis and M. Demetriou, "Feasibility for Orbital Life Extension of a CubeSat in the Lower Thermosphere," *Journal of Spacecraft and Rockets*, vol. 53, no. 5, 2016.
- [53] K. Lemmer, "Propulsion for CubeSats," *Acta Astronautica*, vol. 134, pp. 231-243, 2017.
- [54] National Aeronautics and Space Administration, "Propulsion," 11 March 2020. [Online]. Available: <https://www.nasa.gov>. [Accessed 13 September 2020].
- [55] D. L. Oltrogge and K. Leveque, "An Evaluation of CubeSat Orbital Decay," AIAA/USU, 2011.
- [56] C. Glaser, "Solar Panel Power Tracking," Mouser Electronics, [Online]. Available: <https://www.mouser.com/applications/solar-panel-power-tracking/>. [Accessed 13 10 2020].
- [57] "Bypass Diodes in Solar Panels," Electronics Tutorials, 5 5 2019. [Online]. Available: <https://www.electronics-tutorials.ws/diode/bypass-diodes.html>. [Accessed 13 10 2020].
- [58] Y. Kovo, "State of the Art Small Satellite Technology - Power," NASA, 12 3 2020. [Online]. Available: <https://www.nasa.gov/smallsat-institute/sst-soa/power>. [Accessed 13 10 2020].

- [59] Battery University, "BU-808: How to Prolong Lithium-based Batteries," Battery University, 7 September 2020. [Online]. Available: https://batteryuniversity.com/learn/article/how_to_prolong_lithium_based_batteries. [Accessed 16 March 2021].
- [60] A. Ali, S. A. Khan, M. U. Khan, H. Ali, M. R. Mughal and J. Praks, "Design of Modular Power Management and Attitude Control Subsystems for a Microsatellite," *International Journal of Aerospace Engineering*, vol. 2018, no. CubeSats and Small Satellites, p. 14, 2018.
- [61] "Space Solar Cells," 6 5 2019. [Online]. Available: http://www.azurspace.com/images/006050-01-00_DB_3G30C-Advanced.pdf. [Accessed 10 11 2020].
- [62] "BOOST CONVERTER HAS HIGH EFFICIENCY AT LIGHT LOADS," 9 7 1998. [Online]. Available: <https://www.maximintegrated.com/en/design/technical-documents/app-notes/1/110.html#:~:text=Consuming%20only%20microwatts%20of%20power,efficiency%20at%20low%20load%20currents>. [Accessed 2020 25 11].
- [63] AGI, [Online]. Available: <http://agiweb.force.com/faqs/articles/Keyword/How-to-convert-CAD-models-for-use-in-STK?retURL=%2Ffaqs%2Fapex%2Ffaq&popup=true>. [Accessed 5 10 2020].
- [64] Analytical Graphics, Inc. (AGI), "STK Space Environments Effects Tool (SEET)," 2020. [Online]. Available: <https://solarsystem.nasa.gov/news/1127/10-things-to-know-about-the-ionosphere/>. [Accessed 30 October 2020].
- [65] G. Shirville and B. Klofas, "GENSO: A Global Ground Station Network," 2007.

- [66] NASA, "Near Earth Network (NEN) Services," NASA, 5 12 2017. [Online]. Available: https://www.nasa.gov/directorates/heo/scan/csp/discover_scan_networks/nen_services/. [Accessed 13 10 2020].
- [67] ESA, "Bulletin: Space for Europe," ESA, 1 2 2012. [Online]. Available: <http://esamultimedia.esa.int/multimedia/publications/ESA-Bulletin-149/pageflip.html>. [Accessed 13 10 2020].
- [68] Amazon Web Services, "AWS Ground Station," 2020. [Online]. Available: https://aws.amazon.com/ground-station/?nc2=h_q1_prod_sa. [Accessed 10 2020].
- [69] IQ Spacecom, "SLink-PHY," August 2019. [Online]. Available: https://www.iq-spacecom.com/images/downloads/SLink_SLink-PHY_Datasheet_062019.pdf. [Accessed October 2020].
- [70] IQ Spacecom, "Patch Antenna," 2019. [Online]. Available: https://www.iq-spacecom.com/images/downloads/IQ_spacecom_Patch_Antenna_A4.pdf. [Accessed 10 2020].
- [71] Amazon Web Services, "Global Infrastructure," 2020. [Online]. Available: <https://aws.amazon.com/about-aws/global-infrastructure/?p=ngi&loc=1>. [Accessed 11 2020].
- [72] ISIS, "Full Ground Station Kit S Band," CubeSat Shop, [Online]. Available: <https://www.cubesatshop.com/product/full-ground-station-kit-s-band/>. [Accessed 01 2021].

- [73] Telesat, "Real-Time Latency Rethink Possibilities with Remote Networks," Telesat, July 2020. [Online]. Available: https://www.telesat.com/wp-content/uploads/2020/07/Real-Time-Latency_HW.pdf. [Accessed 17 March 2021].
- [74] H. Curtis, "Satsearch," 21 August 2019. [Online]. Available: <https://blog.satsearch.co/2019-08-21-magnetorquers-an-overview-of-magnetic-torquer-products-available-on-the-global-marketplace-for-space>. [Accessed 20 September 2020].
- [75] N. Bellini, "Magnetic Actuators for Nanosatellite Attitude Control," 9 October 2014. [Online]. Available: <https://amslaurea.unibo.it/7506/>. [Accessed 6 October 2020].
- [76] NewSpace Systems, "NewSpace Systems," [Online]. Available: <https://www.newspacesystems.com/portfolio/sun-sensors/>. [Accessed 29 November 2020].
- [77] Hyperion Technologies, "Satsearch," [Online]. Available: <https://satsearch.co/products/hyperion-technologies-gnss200>. [Accessed 29 November 2020].
- [78] BEI Precision Systems & Space Company, "Satsearch," November 2010. [Online]. Available: <https://satsearch.co/products/bei-precision-systems-model-4311a-linear-servo-accelerometer>. [Accessed 30 November 2020].
- [79] Applied Technology Associates, "Satsearch," 23 March 2017. [Online]. Available: <https://satsearch.co/products/applied-technology-associates-ars-15-mhd-angular-rate-sensor>. [Accessed 29 November 2020].
- [80] Innovative Solutions In Space B.V., "Satsearch," [Online]. Available: <https://satsearch.co/products/isis-isis-on-board-computer>. [Accessed 29 November 2020].

- [81] Satsearch, "Satsearch," [Online]. Available: <https://satsearch.co/products/spacemanic-lodestone-mag-gyro-module>. [Accessed 11 March 2021].
- [82] M. Y. Ovchinnikov, D. S. Roldugin, S. S. Tkachev and V. I. Penkov, "B-dot algorithm steady-state motion performance," *Acta Astronautica*, vol. 146, pp. 66-72, 2018.
- [83] H. D. Blach, "A passive system for determining the attitude of a satellite," *AIAA Journal*, vol. 2, no. 7, 1964.
- [84] A. Gatherer and Z. Manchester, "Magnetorquer-Only Attitude Control of Small Satellites using Trajectory Optimization," in *Proceedings of 2019 AAS/AIAA Astrodynamics Specialist Conference*, 2019.
- [85] R. Wisniewski and M. Blanke, "Three-Axis Satellite Attitude Control Based on Magnetic Torquing," *IFAC Proceedings Volumes*, vol. 29, no. 1, pp. 7618-7623, 1996.
- [86] UCAR Center for Science Education, "The Ionosphere," 2020. [Online]. Available: <https://scied.ucar.edu/ionosphere>. [Accessed 20 November 2020].
- [87] NOAA, "Space Weather Prediction Center: Ionosphere," 2020. [Online]. Available: <https://www.swpc.noaa.gov/phenomena/ionosphere>. [Accessed 2 November 2020].
- [88] K. Groves and S. Radicella, "The Ionosphere: Basic Characteristics and Low-Latitude Structure," in *Boston College Institute for Scientific Research*, Kigali, Rwanda, 2014.
- [89] Electronics Notes, "Ionospheric Layers: D, E, F, F1, F2, Regions," [Online]. Available: <https://www.electronics-notes.com/articles/antennas-propagation/ionospheric/ionospheric-layers-regions-d-e-f1-f2.php>. [Accessed 30 September 2020].

- [90] X. H. Mo, D. H. Zhang, J. Liu, Y. Q. Hao, J. F. Ye, J. S. Qin, W. X. Wei and Z. Xiao, "Morphological Characteristics of Equatorial Ionization Anomaly Crest Over Nanning Region," in *USRI Asia-Pacific Radio Science Conference*, Seoul, South Korea , 2016.
- [91] NOAA, "Space Weather Prediction Center: Ionospheric Scintillation," 2020. [Online]. Available: <https://www.swpc.noaa.gov/phenomena/ionospheric-scintillation>. [Accessed 20 November 2020].
- [92] S. Frazier and L. Tran, "10 Things to Know About the Ionosphere," NASA, 10 December 2019. [Online]. Available: <https://solarsystem.nasa.gov/news/1127/10-things-to-know-about-the-ionosphere/>. [Accessed 20 November 2020].
- [93] R. Braeunig, "Rocket & Space Technology: Atmosphere Properties," [Online]. Available: <http://www.braeunig.us/space/atmos.htm>. [Accessed 1 February 2021].
- [94] NASA Glenn Research Center, "The Drag Equation," 5 May 2015. [Online]. Available: <https://www.grc.nasa.gov/www/k-12/airplane/drageq.html>. [Accessed 1 February 2021].
- [95] T. Mikaelian, "Spacecraft Charging and Hazards to Electronics in Space," York University , 2001.
- [96] R. A. Austin, N. Mahadevan, B. D. Sierawski, G. Karsai, A. F. Witulski and J. Evans, "A CubeSat-payload radiation-reliability assurance case using goal structuring notation," in *2017 Annual Reliability and Maintainability Symposium (RAMS)*, Orlando, Florida, 2017.
- [97] NASA, "Solar Cycle 25 Is Here. NASA, NOAA Scientists Explain What That Means," 15 September 2020. [Online]. Available: <https://www.nasa.gov/press-release/solar-cycle-25-is-here-nasa-noaa-scientists-explain-what-that-means>. [Accessed 1 February 2021].

- [98] Systems Tool Kit (STK), "STK Help: STK SEET Particle Flux," February 2021. [Online]. Available: [https://help.agi.com/stk/#seet/Sat_Basic_ParticleFlux.htm?TocPath= Addon%2520Modules%257Cspace%2520Environment%2520and%2520Effects%2520Tool%2520\(SEET\)%257C_____5](https://help.agi.com/stk/#seet/Sat_Basic_ParticleFlux.htm?TocPath=Addon%2520Modules%257Cspace%2520Environment%2520and%2520Effects%2520Tool%2520(SEET)%257C_____5). [Accessed 9 March 2021].
- [99] Systems Tool Kit (STK), "STK Help: Data Provider Elements: SEET Debris Flux," February 2021. [Online]. Available: https://help.agi.com/stk/#./Subsystems/dataProviders/Content/html/dataProviders/SEET_Debris_Flux.htm. [Accessed 9 March 2021].
- [100] K. E. Boushon, "Thermal analysis and control of small satellites in low Earth orbit," Missouri Institute of Science and Technology, Rolla, 2018.
- [101] N. Bograd, G. Jacobson, P. Kroyack, C. Lopez and J. Peters, "Design and Analysis for CubeSat Missions," Worcester Polytechnic Institute, Worcester, 2018.
- [102] R. Coker, "Thermal Modeling and Testing of the Edison Demonstration of Smallsat Networks Project," in *International Conference on Environmental Systems*, Tuscon, 2014.
- [103] COMSOL Mutiphysics, "Heat Transfer Module User's Guide," 2019.
- [104] WireKinetics CO., LTD, "Ultra Heating Fabric," [Online]. Available: <https://www.adafruit.com/product/1481#description>. [Accessed 8 March 2021].
- [105] S. Maus, C. C. Finaly, C. D. Beggan, T. N. Bondar, A. Chambodut, T. A. Chernova, A. Chulliat, V. P. Golovkov, B. Hamilton, M. Hamoudi, R. Holmhe, G. Hulot, W. Kuang, B. Langlais, V. Lesur, F. J. Lowes, H. Lühr, S. Macmillan, M. Manda, S. McLean, C. Manoj

and Menv, "International Geomagnetic Reference Field: the eleventh generation," *Geophysical Journal International*, vol. 183, no. 3, pp. 1216-1230, 2010.

- [106] NOAA National Centers for Environmental Information, "Magnetic Field Calculators," [Online]. Available: <https://www.ngdc.noaa.gov/geomag/calculators/magcalc.shtml#igrfwmm>. [Accessed 12 October 2020].
- [107] N. Le Renard, A. Nicholas, J. Skaro, L. Thiros and M. Tran, "PolySat Helmholtz Cage," California Polytechnic State University, San Luis Obispo, 2017.
- [108] Kungber, "Kungber DC Power Supply Variable, 30V 10A Adjustable Switching Regulated DC Bench Linear Power Supply with 4-Digits LED Power Display 5V2A USB Output, Coarse and Fine Adjustments with Alligator Leads," Kungber, 2021. [Online]. Available: https://www.amazon.com/Kungber-Adjustable-Switching-Regulated-Adjustments/dp/B08DJ1LP2Y/ref=sr_1_4?dchild=1&keywords=power+supply+computer+controlled&qid=1605217160&sr=8-4. [Accessed 11 March 2021].
- [109] Cytron Technologies, "30A 5-30 V Single Brushed DC Driver," [Online]. Available: <https://www.robotshop.com/en/cytron-30a-5-30v-single-brushed-dc-motor-driver.html>. [Accessed 10 March 2021].
- [110] B. Siefert, "LIS3MDL Triple-axis Magnetometer," Adafruit, 31 October 2020. [Online]. Available: <https://learn.adafruit.com/lis3mdl-triple-axis-magnetometer>. [Accessed 16 November 2020].
- [111] Adafruit, "Adafruit LSM6DSOX + LIS3MDL - Precision 9 DoF IMU - STEMMA QT / Qwiic," Adafruit, 2021. [Online]. Available: <https://www.adafruit.com/product/4517>. [Accessed 11 March 2021].

- [112] Adafruit, "Adafruit INA260 High or Low Side Voltage, Current, Power Sensor," Adafruit, 2021. [Online]. Available: <https://www.adafruit.com/product/4226>. [Accessed 11 March 2021].
- [113] Arduino AG, "Arduino MEGA 2560 REV3," Arduino AG, 2021. [Online]. Available: <https://store.arduino.cc/usa/mega-2560-r3>. [Accessed 9 March 2021].
- [114] Remington Industries, "Magnet Wire, 17 AWG Enameled Copper - 7 Spool Sizes," Remington Industries, 2021. [Online]. Available: <https://www.remingtonindustries.com/magnet-wire/magnet-wire-17-awg-enameled-copper-7-spool-sizes/>. [Accessed 11 March 2021].
- [115] L. Alminde, "Educational Benefits from the AAU-Cubesat Student Satellite Project," Aalborg University, Aalborg, Denmark, 2003.
- [116] J. Praks, A. Kestilä, T. Tikka, H. Leppinen, O. Khurshid and M. Hallikainen, "AALTO-1 Earth Observation Cubesat Mission — Educational Outcomes," in *2015 IEEE International Geoscience and Remote Sensing Symposium (IGARSS)*, Milan, 2015.
- [117] E. Caday-Eames, "Small Box, Big Potential: Boeing engineers think 'outside the cube' in satellite development," *Boeing Frontiers*, p. 20, October 2006.
- [118] Aerojet Rocketdyne, "Aerojet Rocketdyne Signs NASA Contract to Mature Development of MPS-130 CubeSat Modular Propulsion System with Green Propellant," Aerojet Rocketdyne, 11 August 2016. [Online]. Available: <https://www.rocket.com/article/aerojet-rocketdyne-signs-nasa-contract-mature-development-mps-130-cubesat-modular-propulsion-system>. [Accessed 3 October 2020].

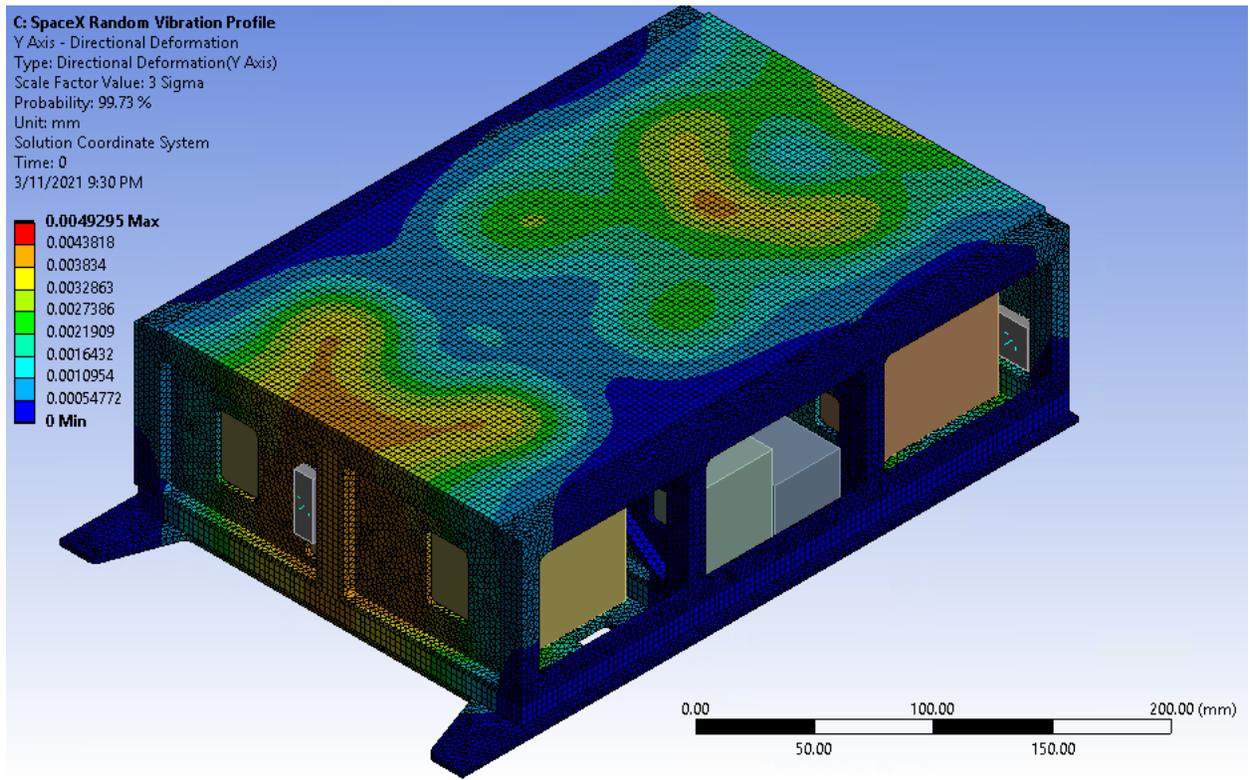
- [119] The Planetary Society, "LightSail: Flight by Light for CubeSats," The Planetary Society, 28 September 2020. [Online]. Available: <https://www.planetary.org/sci-tech/lightsail>. [Accessed 3 October 2020].
- [120] K. Woellert, P. Ehrenfreund, A. Ricco and H. Hertzfeld, "Cubesats: Cost-effective science and technology platforms for emerging and developing nations," *Advances in Space Research*, vol. 47, no. 4, pp. 663-684, 2011.
- [121] R. Hoyt, J. Slostad and R. Twiggs, "The Multi-Application Survivable Tether (MAST) Experiment," Tethers Unlimited, 2003.

Appendices

Appendix A: ANSYS Custom Material Specifications

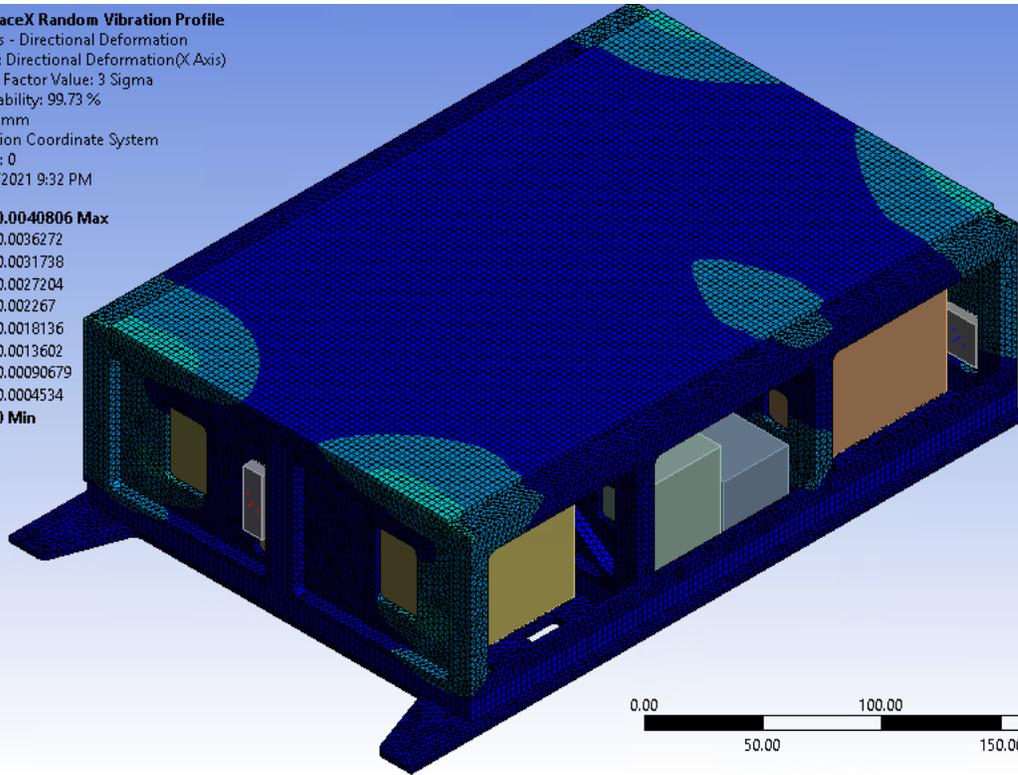
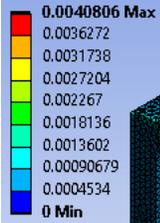
Part	Mass (g)	Effective Density (kg/m ³)
Mini-INMS	960	785.7517025
Transceiver	190	735.0096712
IPBP	252	1388.888889
IPBU	48.7	563.6574074
IPCU	58	559.4135802
IPDU	58	479.4973545
Computer	76	709.3787336
GPS	3	3358.710255
Magnetorquer	196	1334.33154
Linear Accelerometer	170	1623.516278
Angular Velocity Sensor	60	5079.439642
2U Solar Panel	150	2235.053083
6U Solar Panel	300	1846.153846
Patch Antenna	49	3333.333333
Propulsion Stand-in (wet mass)	1638	1259.515571
Sun Sensor	5	2295.684114
Discrete Magnetometer	2	909.0909091

Appendix B: SpaceX Vibration Spectrum Deformation Results



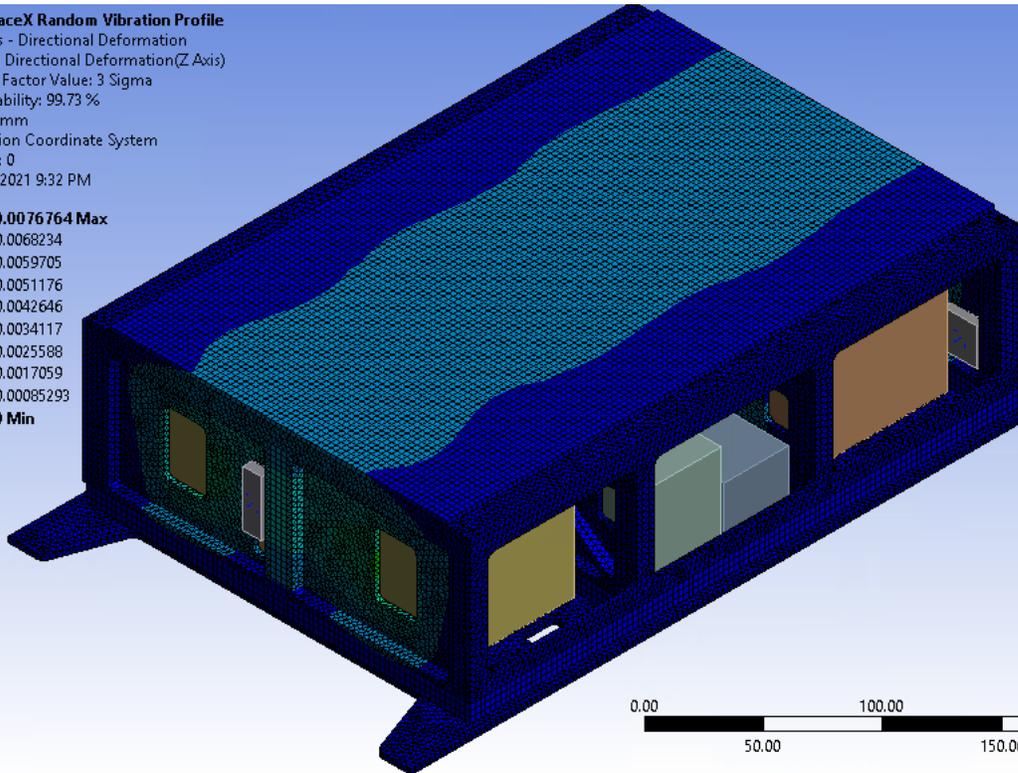
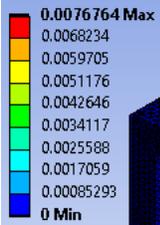
C: SpaceX Random Vibration Profile

X Axis - Directional Deformation
Type: Directional Deformation(X Axis)
Scale Factor Value: 3 Sigma
Probability: 99.73 %
Unit: mm
Solution Coordinate System
Time: 0
3/11/2021 9:32 PM



C: SpaceX Random Vibration Profile

Z Axis - Directional Deformation
Type: Directional Deformation(Z Axis)
Scale Factor Value: 3 Sigma
Probability: 99.73 %
Unit: mm
Solution Coordinate System
Time: 0
3/11/2021 9:32 PM



Appendix C: MATLAB Script for Initial Estimate of ΔV

```
clc; clear all;

alt_i = 625*10^3; %initial altitude (m)
alt_f = 200*10^3; %final altitude (m)
rE = 6371*10^3; %radius of Earth (m)
mu = 3.986*10^14; %m^3/s^2

ra_t = rE + alt_i; %radius of apoapsis of transfer orbit
rp_t = rE + alt_f; %radius of periapsis of transfer orbit
a_t = (ra_t+rp_t)/2; %semi-major axis of transfer orbit
e_t = (a_t-rp_t)/a_t ; %eccentricity of transfer orbit
p_t = a_t*(1-e_t^2); %orbital parameter of transfer orbit
h_t = sqrt(p_t*mu); %specific angular momentum of transfer orbit
ep_t = -mu/(2*a_t); %specific energy of transfer orbit
vt_1 = sqrt(2*(ep_t+mu/ra_t)); %initial velocity of transfer (after
    first impulse)
vt_2 = sqrt(2*(ep_t+mu/rp_t)); %final velocity of transfer (before
    second impulse)

r1 = alt_i + rE; %distance from center of Earth on initial orbit
i1 = 97.8911*pi/180; %initial orbit inclination (rad) %from STK
i2 = 96.3318*pi/180; %final orbit inclination (rad) %from STK
a1 = 7003.14*10^3; %semi-major axis of initial orbit (m) %from STK
ep1 = -mu/(2*a1); %specific energy first orbit
v1 = sqrt(2*(ep1+mu/r1)); %velocity magnitude on initial orbit
delta_i1 = 0; %change in inclination due to first impulse
delta_V1 = sqrt(v1^2+vt_1^2-2*v1*vt_1*cos(delta_i1));

r2 = alt_f + rE; %distance from center of Earth on final orbit
a2 = 6578.14*10^3; %semi-major axis of final orbit (m) %from STK
ep2 = -mu/(2*a2); %specific energy final orbit
v2 = sqrt(2*(ep2+mu/r2)); %velocity magnitude on final orbit
delta_i2 = 0; %change in inclination due to second impulse
delta_V2 = sqrt(v2^2+vt_2^2-2*v2*vt_2*cos(delta_i2));

delta_v_budget = delta_V1 + delta_V2 %m/s
```

Appendix D: MATLAB Script for Determining ΔV Budget with Expanded

Thruster Tank

```
n = 3; %number of tanks
mtank = 1.15*.1; %mass of tank
mdry_thruster = 1.15 + mtank*(n-1); %dry mass of thruster with one
    tank = 1.15
mf_thruster = mdry_thruster;
mp = .086 * n; %propellant mass in one tank = 0.086
mf_sat = 6.0632 + mf_thruster; %dry mass of satellite without
    propulsion system 6.0632kg
mi_thruster = mf_thruster + mp;
mi_sat = mf_sat + mp;
g = 9.8; %acceleration of gravity (m/s^2)
Isp = 800; %specific impulse of BET-1mN (s)
c = g*Isp; %effective exhaust velocity (m/s)
delta_v_capability = c*log(mi_sat/mf_sat) %delta V capability (m/s)
```

Appendix E: STK Access Report Sample

Page 1

11 Mar 2021 01:49:14
 FOR UNFUNDED EDUCATIONAL USE ONLY
 Satellite-250by400km-Transmitter-DLTx1-To-Place-Bahrain, Place-Cape_Town, Place-Columbus, Place-London, Place-Salem, Place-Sao_Paulo, Place-Sydney, Place-Tokyo: Access Summary Report

DLTx1-To-Bahrain

	Access Duration (sec)		Start Time (UTCG)		Stop Time (UTCG)
--	-----		-----		-----
		1	29 Apr 2021 23:40:41.577		29 Apr 2021 23:48:12.0
94	450.518	2	30 Apr 2021 01:11:50.004		30 Apr 2021 01:17:24.1
26	334.122	3	30 Apr 2021 12:15:31.380		30 Apr 2021 12:24:43.5
37	552.156	4	30 Apr 2021 13:45:48.664		30 Apr 2021 13:54:16.6
40	507.976	5	30 Apr 2021 23:56:36.418		1 May 2021 00:04:38.4
57	482.038	6	1 May 2021 01:30:25.006		1 May 2021 01:31:36.6
53	71.647	7	1 May 2021 12:31:30.875		1 May 2021 12:41:19.4
05	588.530	8	1 May 2021 14:02:39.414		1 May 2021 14:09:32.8
06	413.392	9	1 May 2021 22:46:03.991		1 May 2021 22:48:25.1
29	141.138	10	2 May 2021 00:12:49.079		2 May 2021 00:20:53.0
21	483.942	11	2 May 2021 11:20:13.344		2 May 2021 11:24:08.3
93	235.049	12	2 May 2021 12:47:41.217		2 May 2021 12:57:38.4
19	597.202	13	2 May 2021 14:20:05.165		2 May 2021 14:24:06.6
11	241.447	14	2 May 2021 23:00:29.340		2 May 2021 23:06:04.7
47	335.406	15	3 May 2021 00:29:20.035		3 May 2021 00:36:55.1
53	455.118	16	3 May 2021 11:35:01.991		3 May 2021 11:42:10.2
32	428.241	17	3 May 2021 13:04:01.054		3 May 2021 13:13:41.8
12	580.757	18	3 May 2021 23:15:50.706		3 May 2021 23:22:55.2
35	424.529	19	4 May 2021 00:46:13.722		4 May 2021 00:52:40.3
39	386.618	20	4 May 2021 11:50:36.306		4 May 2021 11:59:20.1
24	523.818	21	4 May 2021 13:20:30.804		4 May 2021 13:29:28.9
25	538.121				

	Access Duration (sec)		Start Time (UTCG)	Stop Time (UTCG)
--	-----		-----	-----
		112	26 May 2021 17:25:49.360	26 May 2021 17:33:20.3
58	450.998	113	26 May 2021 18:56:55.862	26 May 2021 19:02:34.4
63	338.601	114	27 May 2021 06:00:26.425	27 May 2021 06:09:35.7
05	549.280	115	27 May 2021 07:30:40.630	27 May 2021 07:39:13.9
69	513.340	116	27 May 2021 17:41:44.232	27 May 2021 17:49:46.6
58	482.426	117	27 May 2021 19:15:19.434	27 May 2021 19:16:58.5
76	99.142	118	28 May 2021 06:16:25.456	28 May 2021 06:26:12.5
48	587.092	119	28 May 2021 07:47:29.366	28 May 2021 07:54:32.6
05	423.239	120	28 May 2021 16:31:08.267	28 May 2021 16:33:37.0
65	148.798	121	28 May 2021 17:57:56.739	28 May 2021 18:06:01.3
81	484.642	122	29 May 2021 05:05:12.393	29 May 2021 05:08:55.0
39	222.646	123	29 May 2021 06:32:35.304	29 May 2021 06:42:32.5
28	597.224	124	29 May 2021 08:04:49.044	29 May 2021 08:09:12.9
84	263.940	125	29 May 2021 16:45:36.523	29 May 2021 16:51:13.7
05	337.182	126	29 May 2021 18:14:27.325	29 May 2021 18:22:03.9
11	456.587	127	30 May 2021 05:19:58.030	30 May 2021 05:27:00.5
57	422.527	128	30 May 2021 06:48:54.510	30 May 2021 06:58:37.0
03	582.493			
Min Duration	9		1 May 2021 18:14:18.153	1 May 2021 18:14:27.0
17	8.864			
Max Duration	86		20 May 2021 07:06:34.302	20 May 2021 07:16:31.6
37	597.335			
Mean Duration	421.696			
Total Duration	53977.089			
Global Statistics				

Min Duration	9		1 May 2021 18:14:18.153	1 May 2021 18:14:27.0
17	8.864			
Max Duration	23		5 May 2021 19:56:16.891	5 May 2021 20:06:17.7
60	600.868			
Mean Duration	427.034			
Total Duration				

Page 44

	Access	Start Time (UTC)	Stop Time (UTC)
	Duration (sec)		
--	-----	-----	-----
	454791.278		

Appendix F: ADCS Component Decision Matrices

Magnetorquer						
Factor	Mass	Power	Thermal	Size	Magnetic Dipole	Total
Weight	4	5	3	5	2	
ISIS Magnetorquer Board	5	5	4	3	3	80
NanoAvionics Magnetorquers MTQ3X	4	4	5	5	5	86
Magnetometer						
Factor	Mass	Power	Thermal	Range	Error	Total
Weight	4	5	2	7	8	
NewSpace Systems NMRM-Bn25o485	3	3	3	5	5	98
Sputnix SX-MAG-04	3	3	4.5	5	3	92.5
Chang Guang Magnetometer	1	2	5	4	5	97

Spacemaniac LODESTONE SM- ADS-AG4	5	5	5	5	3	114
Meisei Electric Magnetic Sensor	2	2	3	5	3	73
Fine Sun Sensor						
Factor	Mass	Power	Thermal	Field of View	Error	Total
Weight	3	4	2	6	8	
CubeSpace CubeSense	5	3	5	5	4	99
NewSpace Systems NFSS-411	4	5	5	4	5	106
Sputnix SX-SSM- 01	2	2	5	2	4.5	72
Coarse Sun Sensor						
Factor	Mass	Power	Thermal	Field of View	Error	Total
Weight	3	4	2	6	8	
NewSpace Systems NCSS-SA05	4	5	3	4.5	5	105

Space Inventor APS Fine Sun Sensor	3	3	4	5	3	83
Hyperion SS200	5	5	5	4	3	93
GPS Receiver						
Factor	Mass	Power	Thermal	Resolution	Size	Total
Weight	4	4	2	5	3	
Hyperion GNSS200	5	5	5	5	5	90
NewSpace Systems CubeSat GPS Receiver	3	3	4	5	3	66
Sputnix SX-NAV- 03	4	2	4	4	4	64
Linear Accelerometer						
Factor	Mass	Power	Thermal	Error	Size	Total
Weight	4	3	3	5	3	
BEI Model 4310	2	5	5	5	3	72
BEI Model 4384	3.5	5	5	4	5	79

BEI Model 4311A	5	5	5	5	5	90
Angular Velocity Sensor						
Factor	Mass	Power	Thermal	Error	Size	Total
Weight	4	3	3	5	3	
Sputnix SX-AVS-01	4	3	5	unknown	3	49
Applied Technologies ARS-15	5	5	unknown	5	5	75
Onboard Computer						
Factor	Mass	Power	Thermal	Processing Speed	Storage	Total
Weight	3	5	3	5	3	
Sputnix SXC-MB-04	5	2	5	5	5	80
ISIS On-board Computer	3	5	4	4	5	81

Appendix G: B-dot Controller MATLAB Script

```
% B dot controller for detumble simulation with STK Attitude Simulator
```

```
function [output] = B_dot_controller(input)
switch input.method
case 'register'
    % Outputs
    % Torque vector
    torque_O = {'ArgumentType', 'Output',...
                'ArgumentName', 'Torque',...
                'Type', 'Parameter',...
                'Name', 'Torque',...
                'BasicType', 'Vector'};

    % Inputs
    % Magnetic field vector
    B_field_I = {'ArgumentType', 'Input',...
                 'ArgumentName', 'Bfield',...
                 'Name', 'MagField(IGRF)',...
                 'Type', 'Vector',...
                 'RefName', 'Body'};

    % Satellite attitude
    att_I = {'ArgumentType', 'Input',...
             'ArgumentName', 'att',...
             'Type', 'Attitude',...
             'Derivative', 'Yes'};

    % Output data to STK
    output = {torque_O,B_field_I,att_I};

case 'compute'
    % Collect input data
    computeData = input.methodData;

    % Controller gain
    k = 1/250;
    % Define B field variable from STK input
    B = computeData.Bfield;
    % B field unit vector
    b = B/norm(B);
    % Angular rates
    omg = computeData.att(5:7);
    % Calculated magnetic dipole moment strength
    m = cross(k/norm(B)*omg,b);
    % Limit maximum magnetic dipole moment strength
    if m(1) > 0.3
        m(1) = 0.3;
    end
    if m(2) > 0.3
        m(2) = 0.3;
    end
    if m(3) > 0.34
        m(3) = 0.34;
    end
end
end
```

```
end
if m(1) < -0.3
    m(1) = -0.3;
end
if m(2) < -0.3
    m(2) = -0.3;
end
if m(3) < -0.34
    m(3) = -0.34;
end

% Torque output cross product of m and B
output.Torque = cross(m,B);
otherwise
    output = [];
end
end
```

Published with MATLAB® R2019a

Appendix H: TRIAD Attitude Determination

```
% TRIAD Attitude Determination

% Input sun vector
% Body
r1 = [0;0;1];
% ECI
R1 = [1;0;0];

% Input B field vector
% Body
r2 = [0;25;0]*10^-6;
% ECI
R2 = [0;0;25]*10^-6;

S = R1/norm(R1);
s = r1/norm(r1);
M = cross(R1,R2)/norm(cross(R1,R2));
m = cross(r1,r2)/norm(cross(r1,r2));

% Attitude matrix
A = [S,M,cross(S,M)]*[s,m,cross(s,m)]'
```

Published with MATLAB® R2019a

Appendix I: Basic PD Attitude Controller

```
function [output] = Basic_PD_controller(input)
switch input.method
case 'register'
    % Outputs
    % Torque vector
    torque_O = {'ArgumentType', 'Output',...
                'ArgumentName', 'Torque',...
                'Type', 'Parameter',...
                'Name', 'Torque',...
                'BasicType', 'Vector'};

    % Inputs
    % Magnetic field vector
    B_field_I = {'ArgumentType', 'Input',...
                 'ArgumentName', 'Bfield',...
                 'Name', 'MagField(IGRF)',...
                 'Type', 'Vector',...
                 'RefName', 'Body'};
    att_I = {'ArgumentType', 'Input',...
             'ArgumentName', 'att',...
             'Type', 'Attitude',...
             'Derivative', 'Yes'};
    erratt_I = {'ArgumentType', 'Input',...
                'ArgumentName', 'erratt',...
                'Type', 'Attitude',...
                'Derivative', 'Yes',...
                'RefName', 'Body',...
                'RefSource', 'Satellite/PerfectPointing'};
    IMtx_I = {'ArgumentType', 'Input',...
              'ArgumentName', 'IMtx',...
              'Type', 'Inertia',...
              'Name', 'Inertia'};

    % Output data to STK
    output = {torque_O,B_field_I,att_I,erratt_I,IMtx_I};

case 'compute'
    % Collect input data
    computeData = input.methodData;

    %quat = computeData.att(1:4)';

    omg = computeData.att(5:7);

    er_quat = computeData.erratt(1:4)';

    IMtx = computeData.IMtx;

    B = computeData.Bfield;

    % Controller gain
    k = 0.001; c = 0.05;
```

```

temp(1) = (k*er_quat(1)*er_quat(4))+(c*omg(1));
temp(2) = (k*er_quat(2)*er_quat(4))+(c*omg(2));
temp(3) = (k*er_quat(3)*er_quat(4))+(c*omg(3));

torque(1) = -IMtx(1)*temp(1)-IMtx(4)*temp(2)-IMtx(7)*temp(3);
torque(2) = -IMtx(2)*temp(1)-IMtx(5)*temp(2)-IMtx(8)*temp(3);
torque(3) = -IMtx(3)*temp(1)-IMtx(6)*temp(2)-IMtx(9)*temp(3);

torque_lim = 1.5*10^-5;

if abs(torque(1)) > torque_lim || abs(torque(2)) > torque_lim
|| abs(torque(3)) > torque_lim
    torque_ratio = torque_lim / max(abs(torque));
    torque = torque_ratio * torque;
end

output.Torque = torque';
otherwise
    output = [];
end
end

```

Published with MATLAB® R2019a

Appendix J: PD Attitude Controller

```
function [output] = PD_controller(input)
switch input.method
case 'register'
    % Outputs
    % Torque vector
    torque_0 = {'ArgumentType', 'Output',...
                'ArgumentName', 'Torque',...
                'Type', 'Parameter',...
                'Name', 'Torque',...
                'BasicType', 'Vector'};

    % Inputs
    % Magnetic field vector
    B_field_I = {'ArgumentType', 'Input',...
                 'ArgumentName', 'Bfield',...
                 'Name', 'MagField(IGRF)',...
                 'Type', 'Vector',...
                 'RefName', 'Body'};
    att_I = {'ArgumentType', 'Input',...
             'ArgumentName', 'att',...
             'Type', 'Attitude',...
             'Derivative', 'Yes'};
    erratt_I = {'ArgumentType', 'Input',...
                'ArgumentName', 'erratt',...
                'Type', 'Attitude',...
                'Derivative', 'Yes',...
                'RefName', 'Body',...
                'RefSource', 'Satellite/PerfectPointing'};
    IMtx_I = {'ArgumentType', 'Input',...
              'ArgumentName', 'IMtx',...
              'Type', 'Inertia',...
              'Name', 'Inertia'};

    % Output data to STK
    output = {torque_0,B_field_I,att_I,erratt_I,IMtx_I};

case 'compute'
    % Collect input data
    computeData = input.methodData;

    %quat = computeData.att(1:4)';

    omg = computeData.att(5:7);

    er_quat = computeData.erratt(1:4)';

    IMtx = computeData.IMtx;

    B = computeData.Bfield;

    % Controller gain
    k = 0.001; c = 0.05;
```

```

temp(1) = (k*er_quat(1)*er_quat(4))+(c*omg(1));
temp(2) = (k*er_quat(2)*er_quat(4))+(c*omg(2));
temp(3) = (k*er_quat(3)*er_quat(4))+(c*omg(3));

d_torque(1) = -IMtx(1)*temp(1)-IMtx(4)*temp(2)-
IMtx(7)*temp(3);
d_torque(2) = -IMtx(2)*temp(1)-IMtx(5)*temp(2)-
IMtx(8)*temp(3);
d_torque(3) = -IMtx(3)*temp(1)-IMtx(6)*temp(2)-
IMtx(9)*temp(3);

for lam = -0.2:0.01:0.2
    m = 1/norm(B)^2 * cross(B,d_torque') + lam * B * 10^5;
    if abs(m(1)) < 0.3 && abs(m(2)) < 0.3 && abs(m(3)) < 0.34
        break
    else
        m = 1/norm(B)^2 * cross(B,d_torque');
        m = 0.3 * m / max(abs(m));
    end
end

torque = cross(m,B);

output.Torque = torque';
otherwise
    output = [];
end
end

```

Published with MATLAB® R2019a

Appendix K: STK to COMSOL conversion MATLAB code

```
%%Preprocessing
clc
clear
%%Load in the .csv's from STK

eLEO4UCubeSatSolarIntensity=csvread('eLEO6UCubeSatSolarIntensity.csv',2,1);

eLEO4UCubeSatSunVectorsFixed=csvread('eLEO6UCubeSatSunVectorsFixed.csv',2,1);

%%Processing Solar Intensity

%Name of SI table
Name_SI =eLEO4UCubeSatSolarIntensity/100;

SI=Name_SI;

%%Processing Sun Vectors

%Name of SV table
Name_SV=eLEO4UCubeSatSunVectorsFixed;

%Create unit vectors

magnitude=sqrt(Name_SV(:,1).^2+Name_SV(:,2).^2+Name_SV(:,3).^2);

Name_SVxkm=Name_SV(:,1)./magnitude;
Name_SVykm=Name_SV(:,2)./magnitude;
Name_SVzkm=Name_SV(:,3)./magnitude;

%Translate into Body-Fixed Axis in COMSOL Model

xu=Name_SVykm;
yu=-Name_SVxkm;
zu=Name_SVzkm;

%Labled as
VarNames= {'TimeEpSec','posx','negx','posy','negy','posz','negz'};
Suv=[xu yu zu];
Fn_u=[1 0 0; -1 0 0; 0 1 0; 0 -1 0; 0 0 1; 0 0 -1];
num=length(Suv);
cos_theta= zeros(1,6);
cos_theta_all=zeros(num,6);
for k=1:num
    for face=1:6
        cos_theta(k,face)= dot(Fn_u(face,:),Suv(k,:));
        if cos_theta(k,face)>0
            cos_theta(k,face)=cos_theta(k,face);
        else
            cos_theta(k,face)=0;
        end
    end
end
end
```

```
%Table Variables and Table
TimeEpSec=Name_SV(:,1);
posx=cos_theta(:,1);
negx=cos_theta(:,2);
posy=cos_theta(:,3);
negy=cos_theta(:,4);
posz=cos_theta(:,5);
negz=cos_theta(:,6);

cos_theta_xyz=table(TimeEpSec, posx,negx,posy,negy,posz,negz);
%%Save tables as a csv file

SI2=table(eLEO4UCubeSatSolarIntensity(:,1),SI);
writetable(SI2, 'SI.csv')
writetable(cos_theta_xyz, 'cos_theta_xyz.csv')
```

Appendix L: MATLAB Script for Plotting ADCS Helmholtz Cage Magnetic

Field

```
% Bz plot
clc; clear all;
syms z

mu0 = 4*pi*10^-7;
n = %value;
I = %value;
a = %value;
gam = 0.5445;
h = gam*2*a;

Bz = 10^4 * mu0*n*I/pi * (2*a^2/((a^2+(z-h/2)^2)*sqrt(2*a^2+(z-
h/2)^2))+2*a^2/((a^2+(z+h/2)^2)*sqrt(2*a^2+(z+h/2)^2)));
Bz0 = 10^4 * 2*mu0*n*I/(a*pi) * 2/((1+gam^2)*sqrt(2+gam^2))

Perc = 100 + 100*(Bz-Bz0)/Bz0;

figure(1)
fplot(Bz, [-h/2, h/2])
xlabel('z Position (m)')
ylabel('Bz Field (G)')

figure(2)
fplot(Perc, [-h/2, h/2])
hold on
fplot(99, [-h/2, h/2])
hold on
fplot(96, [-h/2, h/2])
xlabel('z Position (m)')
ylabel('Percent Bz0')
```

Published with MATLAB® R2019a

Appendix M: MATLAB Script for Determining Viable Coil Configurations

```
% Size, loop, current analysis
clear all; clc
mu0 = 4*pi*10^-7;
gam = 0.5445;
pass = [];
Bz0pass = [];
% Wire gauge
R_m = 16.60992/1000; % 17
%R_m = 20.9428/1000; % 18
%R_m = 26.40728/1000; % 19
%R_m = 33.292/1000; % 20

for a = 0.825:0.025:0.875 % Coil half length range
    for I = 3:0.05:4.75 % Current range
        for n = 15:1:60 % Wire turn range
            Bz0 = 10^4 * 2*mu0*n*I/(a*pi) * 2/
((1+gam^2)*sqrt(2+gam^2));
            if Bz0 >= 1.5 && Bz0 <= 1.52 && I*8*a*n*R_m <= 30 &&
16*n*a <= 480
                pass = [pass; a, I, n];
                Bz0pass = [Bz0pass; Bz0];
            end
        end
    end
end

end
Volt = 8*prod(pass') * R_m;
Pw = 2 * Volt .* pass(:,2)';
len = 16*pass(:,1).*pass(:,3);
figure(1)
plot3(pass(:,1), pass(:,2), pass(:,3))
figure(2)
plot(Pw)
xlabel('Pass counter')
ylabel('Power (W)')
figure(3)
plot(len)
xlabel('Pass counter')
ylabel('Wire Length (m)')
figure(4)
plot(Volt)
xlabel('Pass counter')
ylabel('Voltage (V)')
figure(5)
plot(pass(:,1))
xlabel('Pass counter')
ylabel('Side half length (m)')
```

Appendix N: Helmholtz Cage Instruction Manual

Helmholtz Cage Instruction Manual

Recommended 2-3 people for assembly

Assembly: Outside-In Method

Frame Pairs:

NO and JOE = Inner vertical frame (X axis)

FRED and TED = Outside vertical frame (Y axis)

SMOL and PAUL = inner horizontal frame (Z axis)

NOTE: Make sure to orient the coils so the plugs are closest to the power supply.

NOTE: For each pair make sure the arrows on the coil frame go the same direction

NOTE: Each mounting bracket has two different sides, one with holes 5/8" from the edge of the bracket, and one with holes 3/8" from the edge of the bracket. Make sure to mount using correct side of the brackets so the holes align correctly.

Frame Assembly:

1. Attach brackets facing inwards on FRED and TED with pins.
2. Attach brackets facing inward on NO and JOE with pins, make sure to use the holes furthest from the corners.

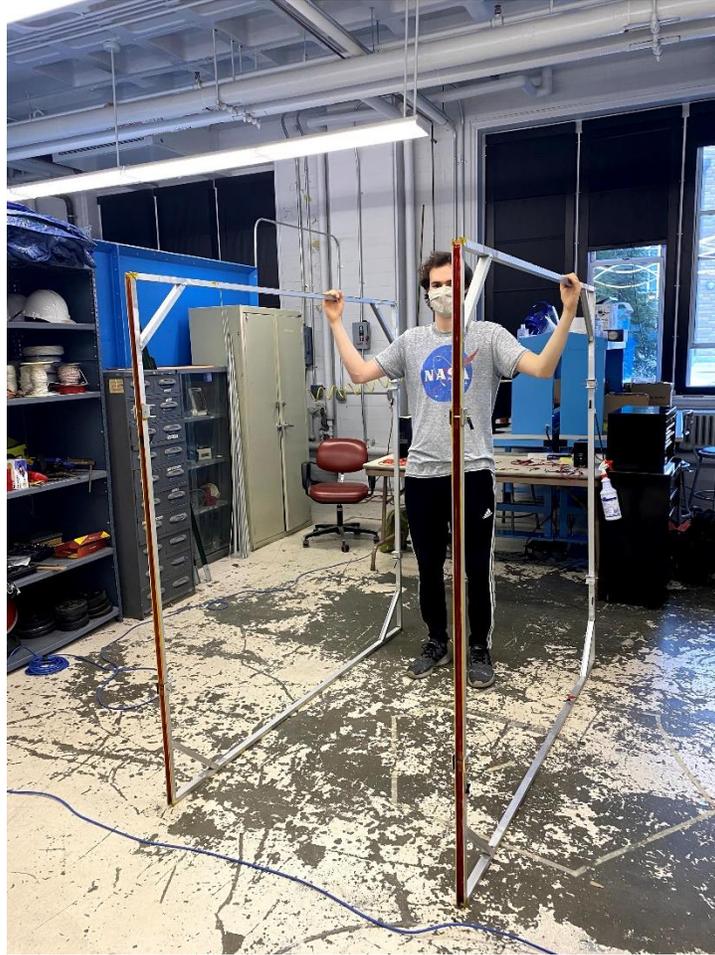


Figure 101: NO and JOE

3. Stand NO and JOE up with the brackets on the two vertical sides.
4. Slot in either PAUL or SMOL between the bottom brackets of NO and JOE, secure with pins. Note: One bracket connection for SMOL has an 'S' on it and matches a customized bracket also labeled with an 'S'.

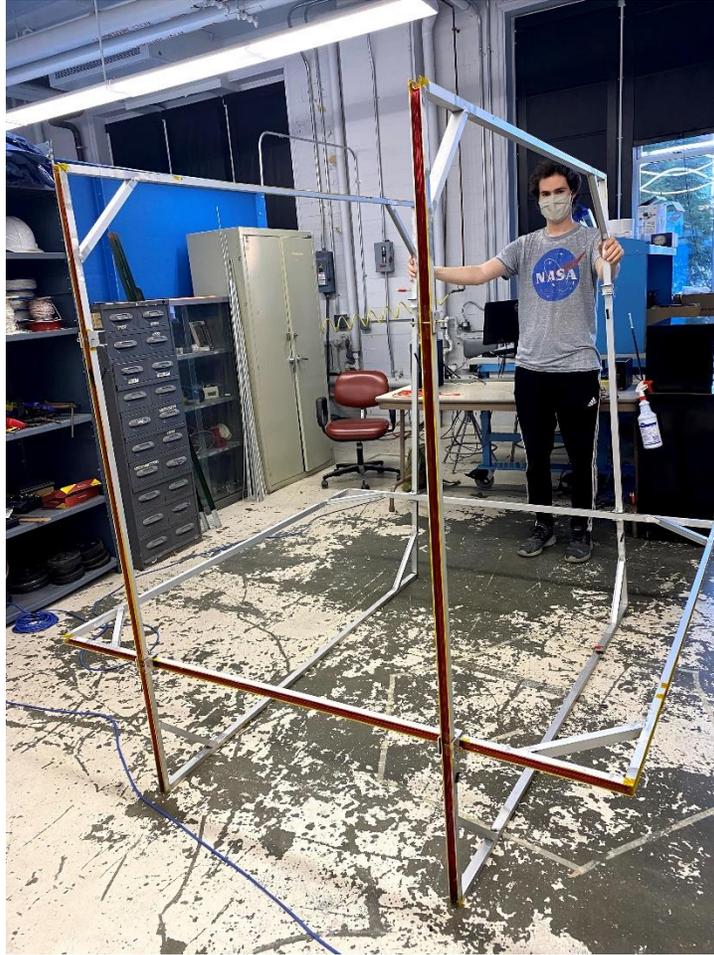


Figure 102: PAUL or SMOL connected to NO and JOE

5. Slot in either PAUL or SMOL (whichever is left) between the top brackets of NO and JOE, secure with pins.



Figure 103: SMOL and PAUL connected to NO and JOE

6. Stand FRED or TED up with the brackets on the top and bottom.
7. Slightly lift the already connected frame and slide FRED or TED onto NO and JOE, secure with pins.

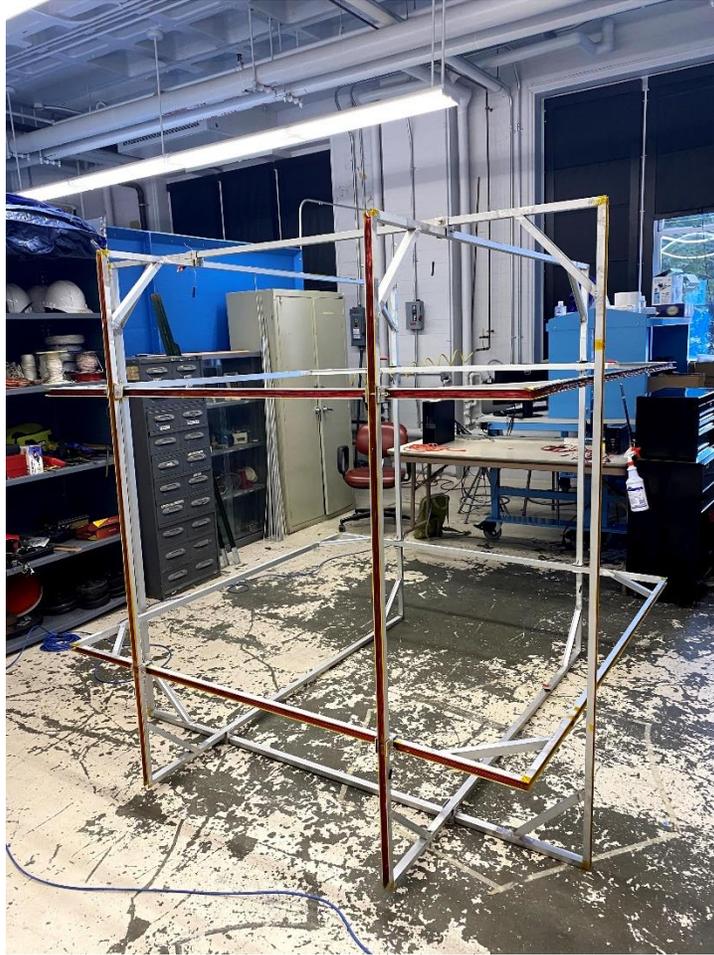


Figure 104: FRED or TED added to frame

8. Repeat steps 6 and 7 for the remaining coil of the pair FRED and TED.

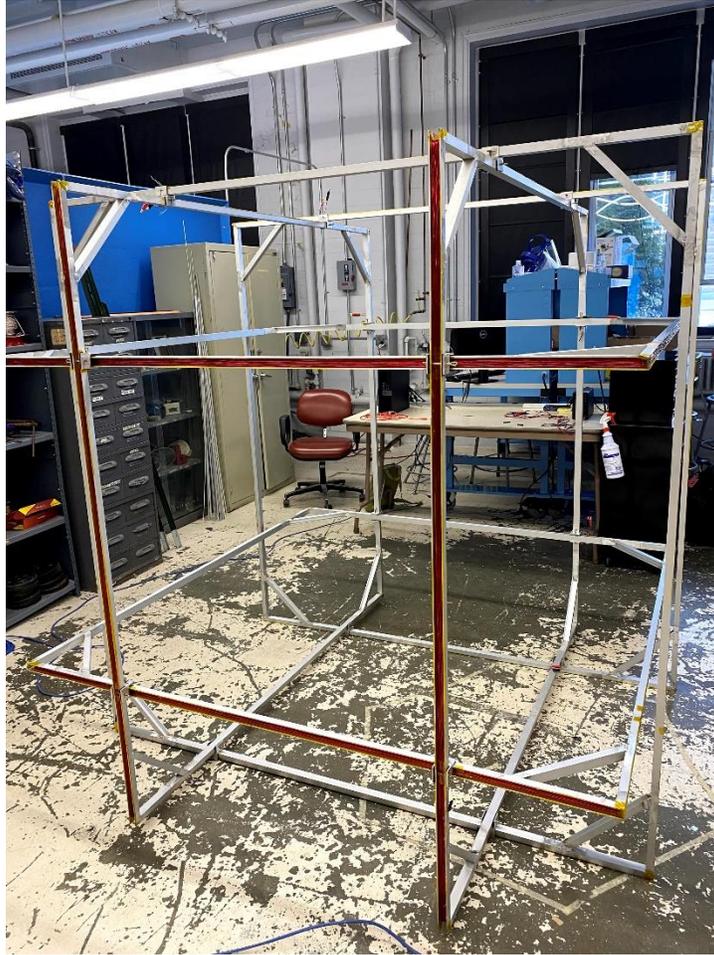


Figure 105: Full frame assembly

9. Slot the loose rubber pieces into the spaces of the brackets between the coils.



Figure 106: Bracket and rubber pieces

Connection Set-Up:

1. Connect labeled banana plug cables to corresponding coil plugs.
2. Connect the other ends of the cables for each coil pair to their corresponding splitter, based on the previously stated criteria.

NOTE: If coil arrow directions are opposite when assembled, attach one side as red-red and black-black and the other side as red-black and black-red.

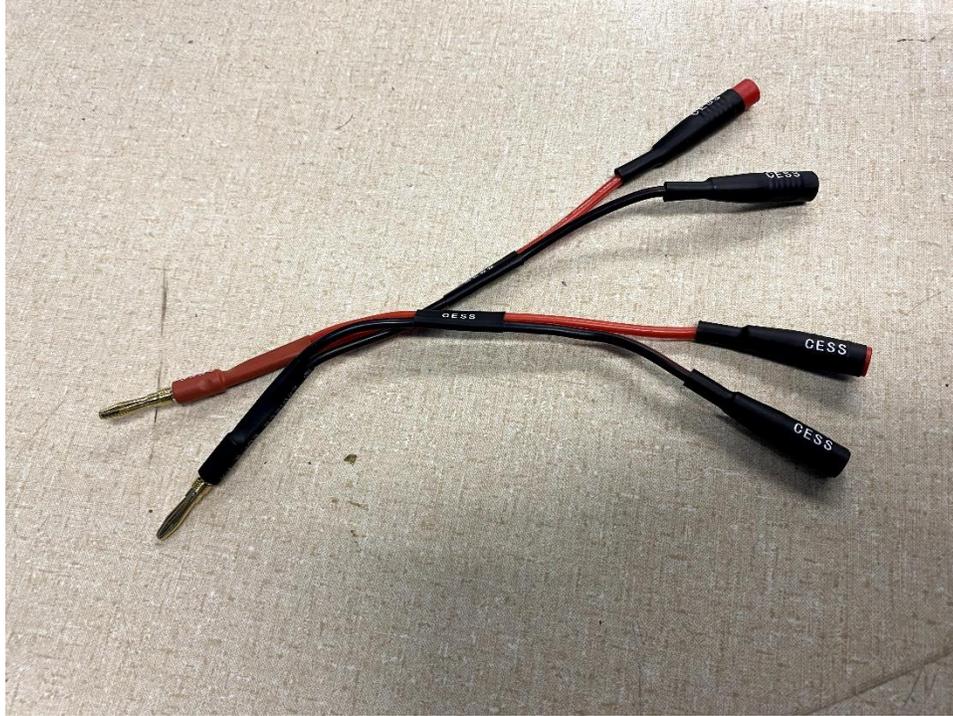


Figure 107: Splitter

3. Red (terminal A) Splitter connects to current sensor via a piece of wire with a female connector.
 - a. Current sensor connected to motor controller via piece of wire.
4. Black (terminal B) piece of splitter connects directly to the motor controller via piece of wire with a female connector.
5. Motor controller connects directly to the power supply via short wires.
 - a. Red is positive and black is negative.

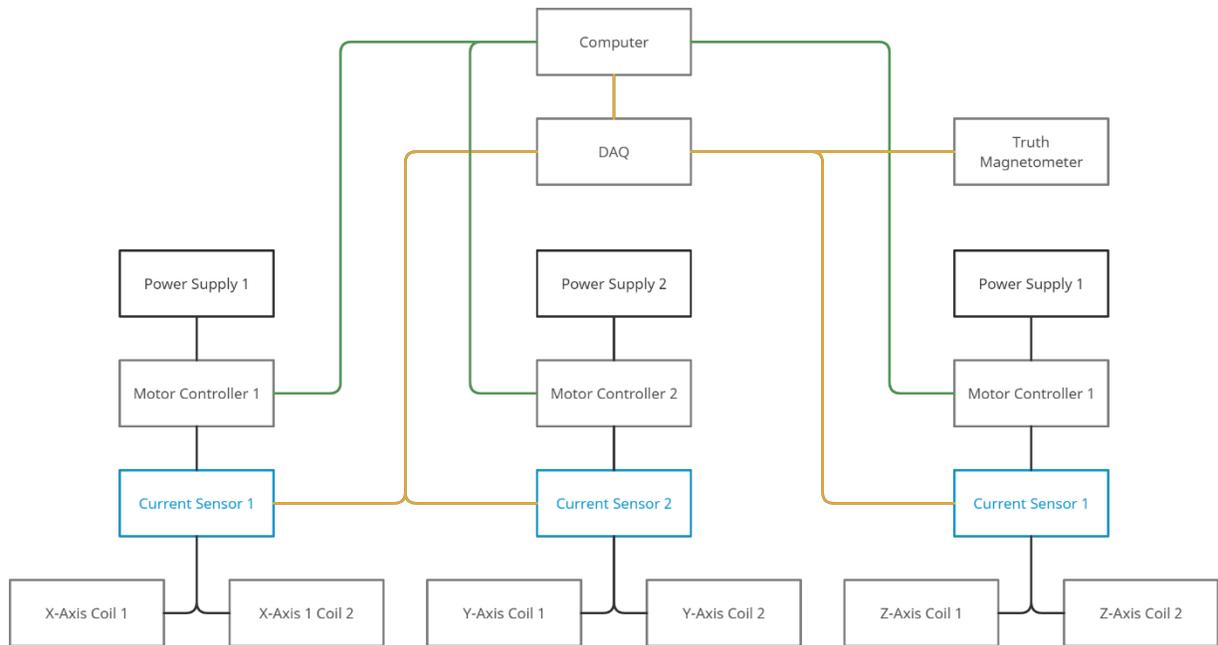
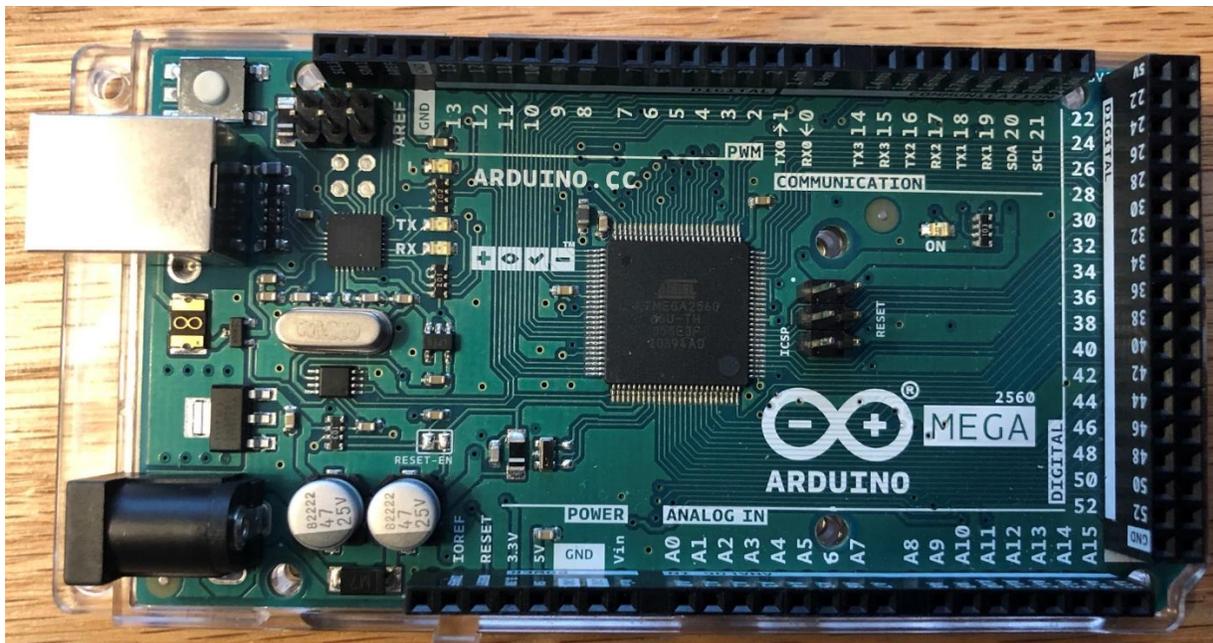


Figure 108: Wiring Diagram

Note: The computer and DAQ are combined in the Arduino and the IMU and magnetometer are combined.

Arduino

An Arduino MEGA 2560 is used for this project. The Arduino can be connected to a computer using a USB-B cable. It can be programmed using the Arduino IDE, which is free, open-source, and available at <https://www.arduino.cc/en/software>. Adafruit libraries are required to use the sensors. These libraries can be downloaded directly from within the Arduino IDE, under Tools->Manage Libraries, and by searching for the Adafruit sensor name. A picture of the Arduino MEGA is below:



The relevant connections are as follows: USB-B connector: this is used to interface with the computer. 3.3V, 5V, and GND pins: these pins are used to provide power and ground to the sensors. SDA (20) and SCL (21): these pins are used for I2C data transfer from the sensors (discussed below). Do not use the alternate SDA1 and SCL1 pins. Pins 2-13: PWM pins. These pins are used to transfer the PWM signal to the motor controllers. Which pin is used is easily

defined in the code. Digital pins 22-53: general purpose digital pins. These pins can be used for the PWM direction signal and defined in the code.

Sensor Connections, Wiring, and Code

CONNECT SENSORS WITH THE ARDUINO UNPLUGGED. Each sensor communicates using the I2C data transmission protocol. Each sensor requires 4 wires: power input, at either 3.3 or 5V (both supplied by Arduino), and two I2C lines: SDA (I2C data) and SCL (I2C clock), each connected to the corresponding pins on the Arduino (20 and 21). The I2C pins can be connected in series, simplifying wiring. Each sensor must have a unique I2C address, leading to issues with multiple of the same sensor, such as in the current sensors. It should also be noted that despite being on the same physical chip, the IMU and integrated magnetometer are considered separate I2C devices with different addresses. To get around this issue the sensors have contact pads which can either be bridged with solder, or a pin which can be directly connected to Vin, to change the I2C address. A table describing these addresses is below:

Sensor	Pads Bridged	I2C Address
Current Sensor 1	None	0x40 (default)
Current Sensor 2	A0	0x41
Current Sensor 3	A1	0x44
Magnetometer 1 (IMU integrated)	ADM	0x1E
Magnetometer 2	None	0x1C (default)
IMU	None	0x6A (default)

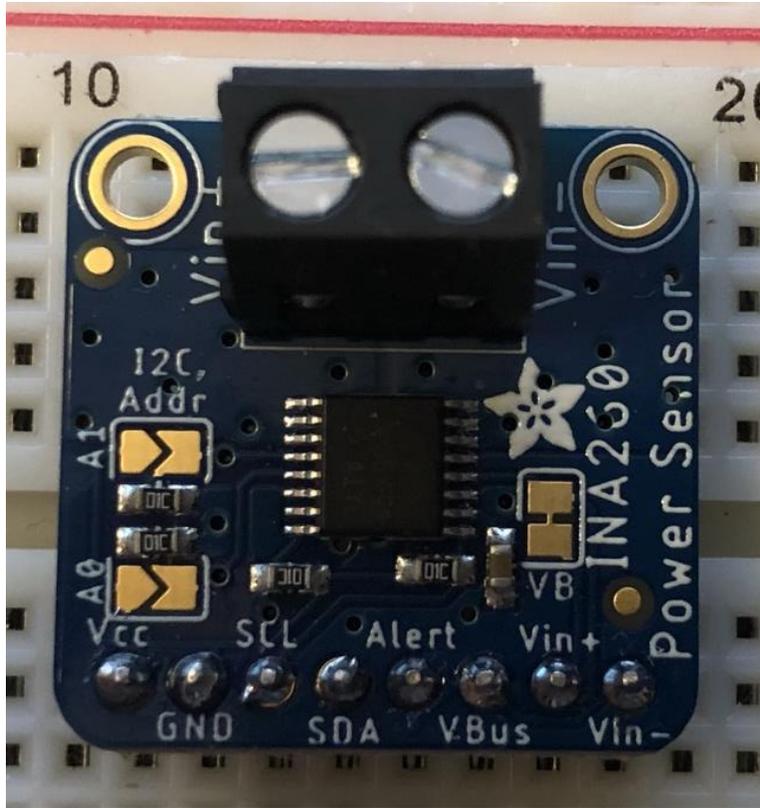
To access the sensors, the relevant Adafruit libraries must be downloaded using the Arduino IDE library manager. Code was adapted from the sensor examples to create the serial connection and access the sensors, attached and annotated in Appendix 1. It should be noted that each sensor object needs to have a unique name and a unique I2C address. If the default address is used, i.e. no pads are bridged, the address does not need to be passed in when beginning the I2C connection.

These sensor inputs can be read using the Arduino IDE serial monitor and plotter, or by using the Python program attached and annotated in Appendix 2, which also writes the data to a specified file. These two methods cannot be used at the same time.

The Python program requires the PySerial library. It should be noted that it overwrites the selected file each time it is restarted, though this could easily be altered as described in the code comments.

Current Sensors

CONNECT SENSORS WITH THE ARDUINO UNPLUGGED. A picture of the current sensor is below:



The current sensor has 8 pins and two screw terminals. The table below shows what they should be connected to:

Pin 1	Pin 2	Pin 3	Pin 4	Pin 5
Vcc	GND	SCL	SDA	Alert
3.3 V or 5 V power from Arduino 3.3 V or 5V	Ground from Arduino Ground	I2C clock from Arduino pin 21: SCL (not SCL1)	I2C data from Arduino pin 20: SDA (not SDA1)	Not Required

Pin 6	Pin 7	Pin 8	Term 1	Term 2
VBus	Vin+	Vin-	Vin+	Vin-

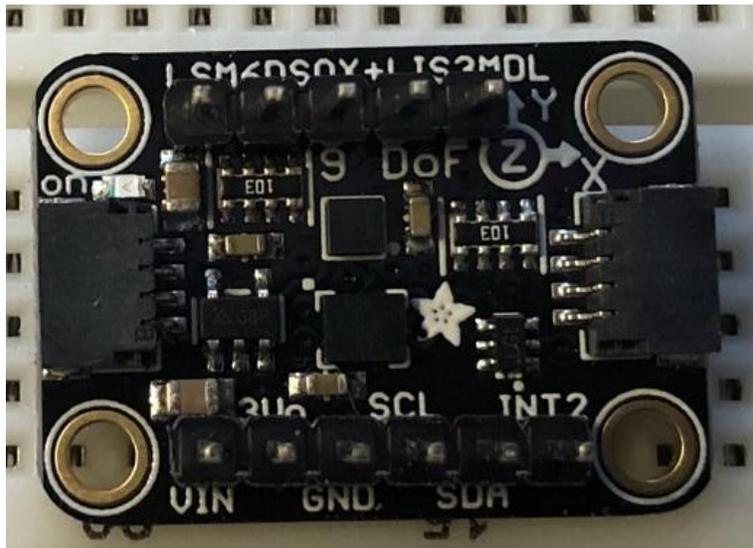
Not required	Not required, can be used instead of Term 1 for low loads, not 5A though.	Not required, can be used instead of Term 2 for low loads, not 5A though.,	Positive line from motor controller	Negative line from motor controller
--------------	---	--	-------------------------------------	-------------------------------------

The current sensor Vin+ and Vin- terminals should be connected inline between the motor controller and banana splitter.

Remember that the three current sensors all have different addresses as described above.

IMU

CONNECT SENSORS WITH THE ARDUINO UNPLUGGED. A picture of the IMU is below:



The IMU pins are attached upside down. Be careful that the pins are being connected correctly.

The IMU has 11 pins.

6-pin row

Pin 1	Pin 2	Pin 3	Pin 4	Pin 5	Pin 6
VIN	3Vo	GND	SCL	SDA	INT2
3.3 V or 5 V power from Arduino 3.3 V or 5V	Not required	Ground from Arduino Ground	I2C clock from Arduino pin 21: SCL (not SCL1)	I2C data from Arduino pin 20: SDA (not SDA1)	Not required

5-pin row

Pin 1	Pin 2	Pin 3	Pin 4	Pin 5
ADAG	ADM	DRDY	INTM	INT1
Not required (used to change IMU address. But there is only one IMU).	Not required (used to change magnetometer address. But the pad is already bridged).	Not required	Not required	Not required

Magnetometer

CONNECT SENSORS WITH THE ARDUINO UNPLUGGED. A picture of the standalone magnetometer is below:

The magnetometer has 10 pins. The table below shows what they should be connected to:

6-pin row

Pin 1	Pin 2	Pin 3	Pin 4	Pin 5	Pin 6
VIN	3Vo	GND	SCL	SDA	INT2
3.3 V or 5 V power from Arduino 3.3 V or 5V	Not required	Ground from Arduino Ground	I2C clock from Arduino pin 21: SCL (not SCL1)	I2C data from Arduino pin 20: SDA (not SDA1)	Not required

4-pin row

Pin 1	Pin 2	Pin 3	Pin 4
GND	DRDY	DO	CS
Not required (extra ground)	Not required	Not required	Not required

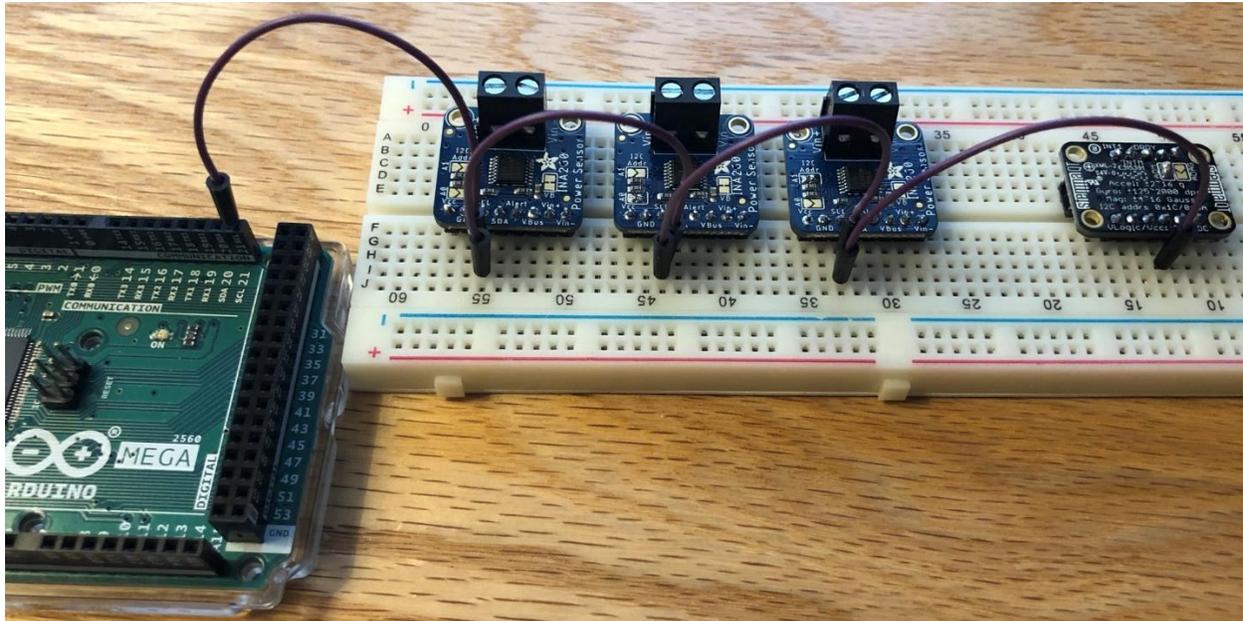
Remember that the two magnetometers (standalone and IMU integrated) have different addresses as described above.

Connecting multiple I2C devices

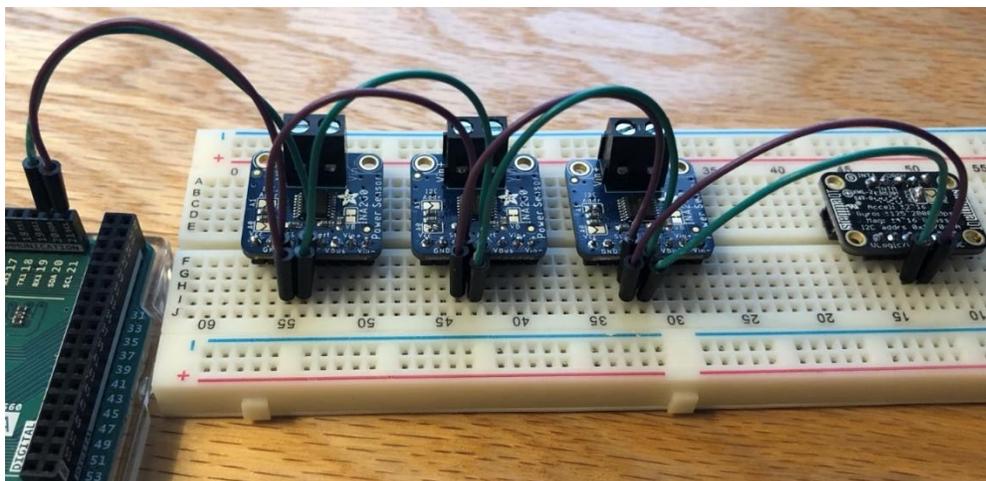
CONNECT SENSORS WITH THE ARDUINO UNPLUGGED. Note that if the project breadboard is available, this section can be skipped as the breadboard takes care of the connections for the user.

Multiple I2C devices can be easily connected. While each device requires its own power and ground connections to the Arduino, I2C Clock (SCL) and I2C Data (SDA) can be connected

in series. An example of this is below, using purple wires to connect each SCL terminal to the Arduino SCL port (power and ground not shown).

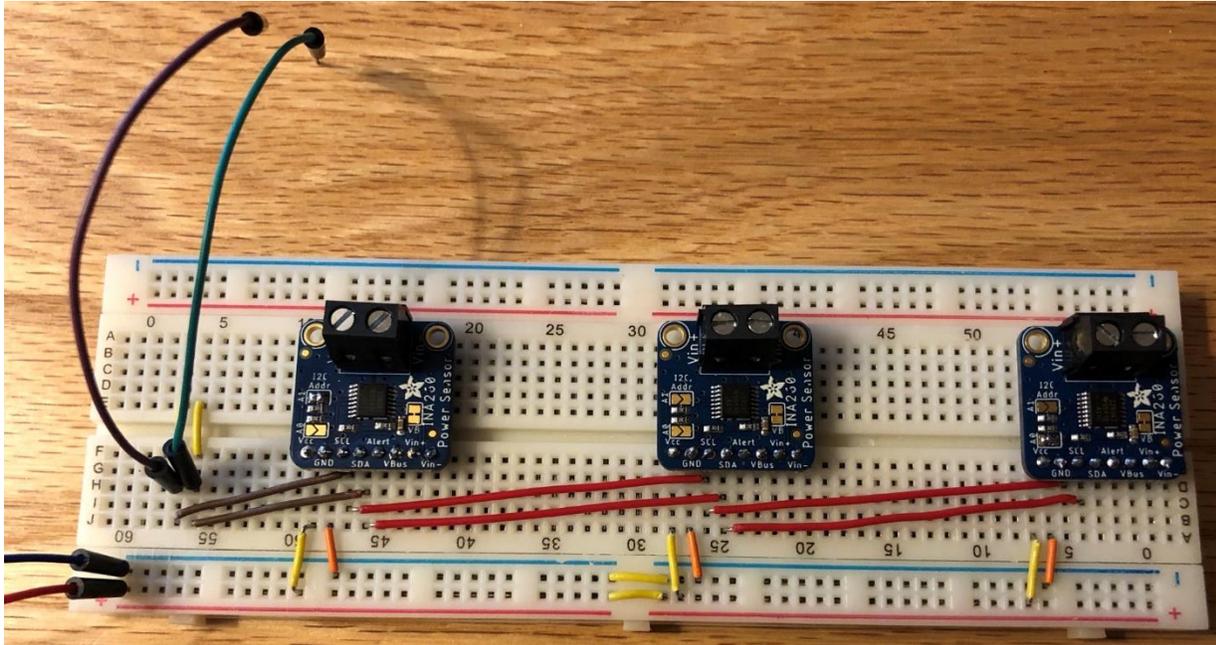


A further example is below, showing purple for I2C Clock (SCL) and green for I2C Data (SDA) (power and ground not shown).

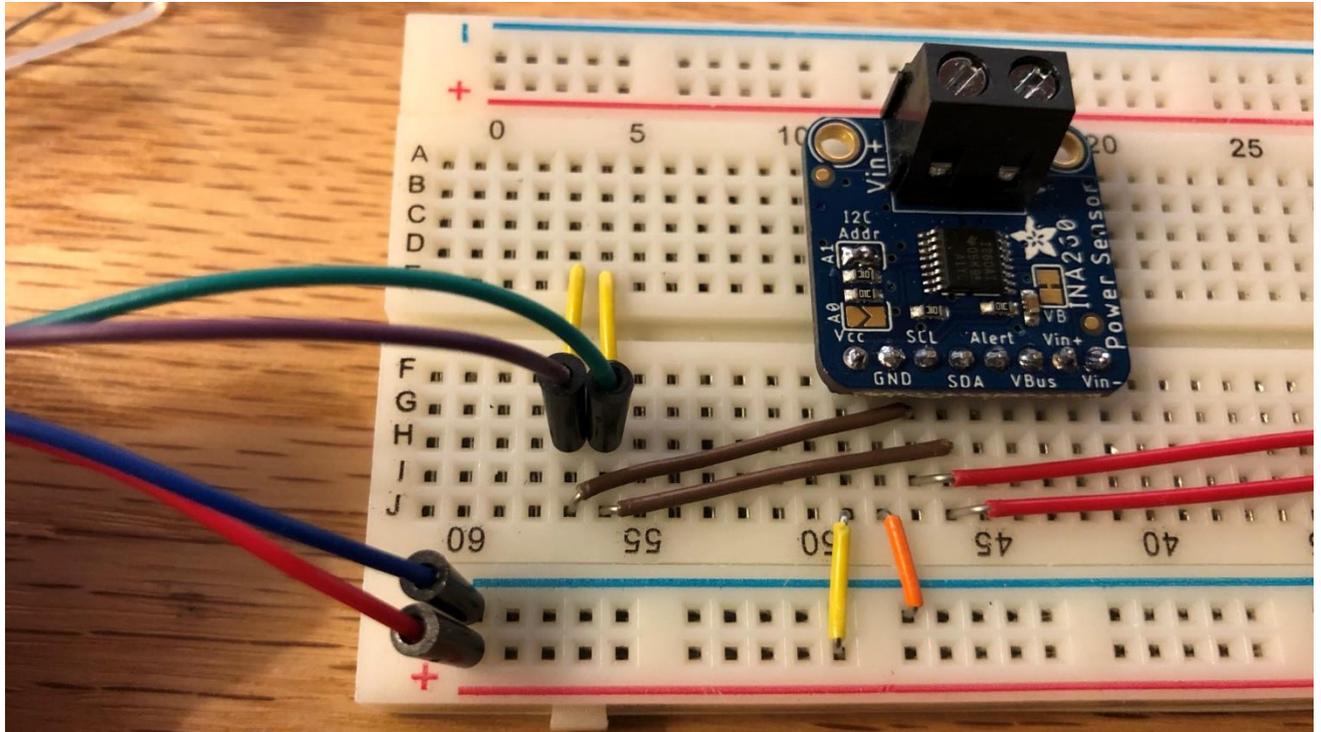


Breadboard

CONNECT SENSORS WITH THE ARDUINO UNPLUGGED, DO NOT FORCE CONNECTIONS INTO THE BREADBOARD. There is a breadboard which helps manage all the connections and completes the previous step without additional user intervention. A picture is below:



Also see the close up of the left side:



Connect the bottom red rail to Arduino power (3.3 or 5 V). Connect the blue rail to Arduino ground. Connect the first column with a wire in it (column 57, purple wire in picture, farthest column to the left with a wire) to Arduino SCL (I2C Clock). Connect the second column with a wire in it (column 56, green wire in picture, second farthest to the left with a wire) to Arduino SDA (I2C Data). This will allow use of the three current sensors. To use the magnetometers and IMU as well, use the ethernet cable to connect each line to the appropriate thing. The power/ground wire pair can be placed directly on the power rail. The I2C pair can be placed horizontally, with the I2C Clock in the I2C Clock column, and the I2C Data in the I2C Data column. Be very careful which is which! See the Ethernet cable section for further details. Because the upper half (A-E) and lower half (F-J) are bridged, the entire column can be used for I2C connections IN THE CORRECT ORIENTATION.

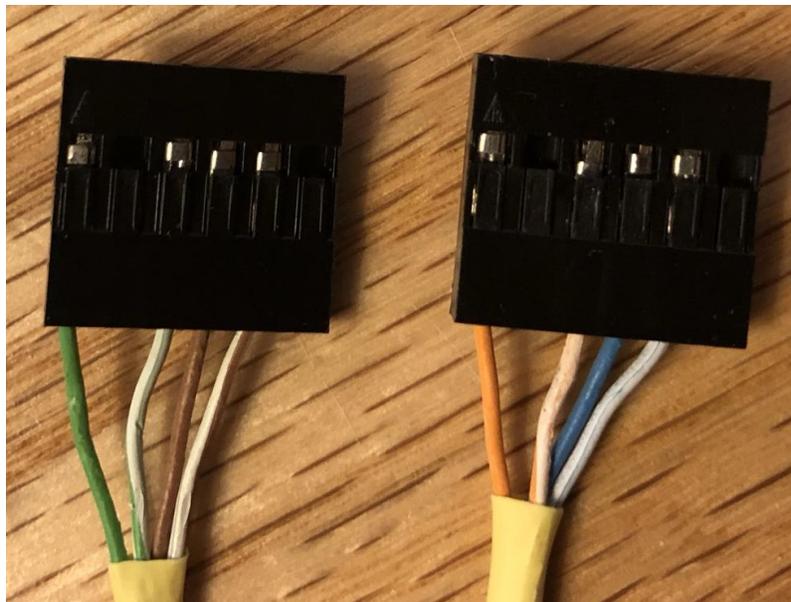
Note that the breadboard build quality is poor. Do not try to force connections. Try a different orientation or position instead.

Ethernet Cable (IMU and Magnetometers)

CONNECT SENSORS WITH THE ARDUINO UNPLUGGED AND DO NOT FORCE CONNECTIONS INTO THE BREADBOARD. Also note that the ethernet cable end connectors can be somewhat fragile and may require reseating the metal within the plastic.

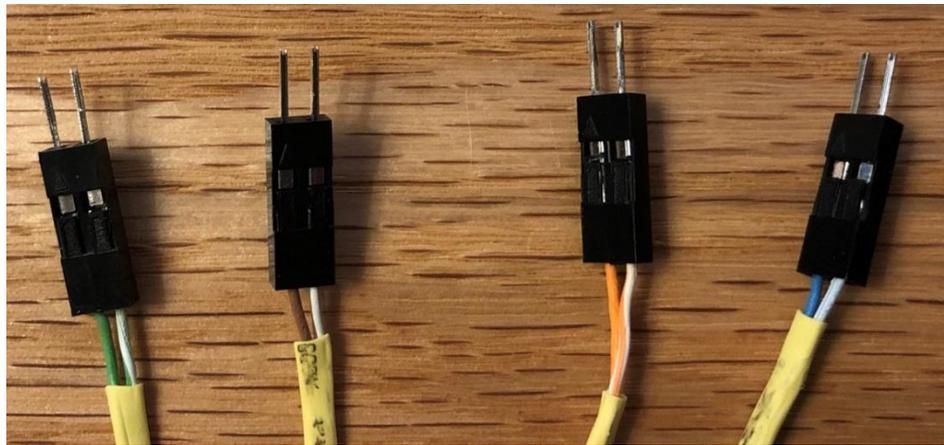
An old Cat5 ethernet cable was used for connectivity between the test article, magnetometer, and Arduino. The ethernet cable has 4 twisted pairs for 8 total wires, exactly what was necessary. BEFORE USE MAKE SURE THE ETHERNET CABLE IS ATTACHED SECURELY TO THE BASE BOARD OR TABLE. The end of the ethernet cable at the Arduino must be solidly attached to the base board or a table. If not, the cable is stiff enough to come unplugged, or move the breadboard and Arduino.

One end is intended to accept the IMU and Magnetometer, and has two 6-pin receptacles, with 4 pins filled:



The first solid color wire (green or orange) is for power (Vcc pin). The subsequent gap IS INTENDED. The next white/previous color wire (white/green or white/orange) wire is for ground. The next solid color wire (brown or blue) is for I2C Clock (SCL pin). The final wire (white/brown or white/blue) is for I2C Data (SDA pin). MAKE SURE THE CONNECTOR IS ATTACHED THE CORRECT WAY TO THE SENSOR.

The other end is for the breadboard and has four 2-pin pin sets:



The wire colors correspond directly with the other end. Thus, connect the green/white and orange/white pairs directly to the power rail, with the solid color on positive. Make sure the connection is stable and secure. The other two wire pairs are for I2C Clock and I2C data. Connect the SOLID color (brown or blue) to the I2C Clock column. Connect the WHITE color (white/brown or white/blue) to the I2C Data column. As the columns are directly adjacent, this is simply. Just make sure the jumper is the right way around.

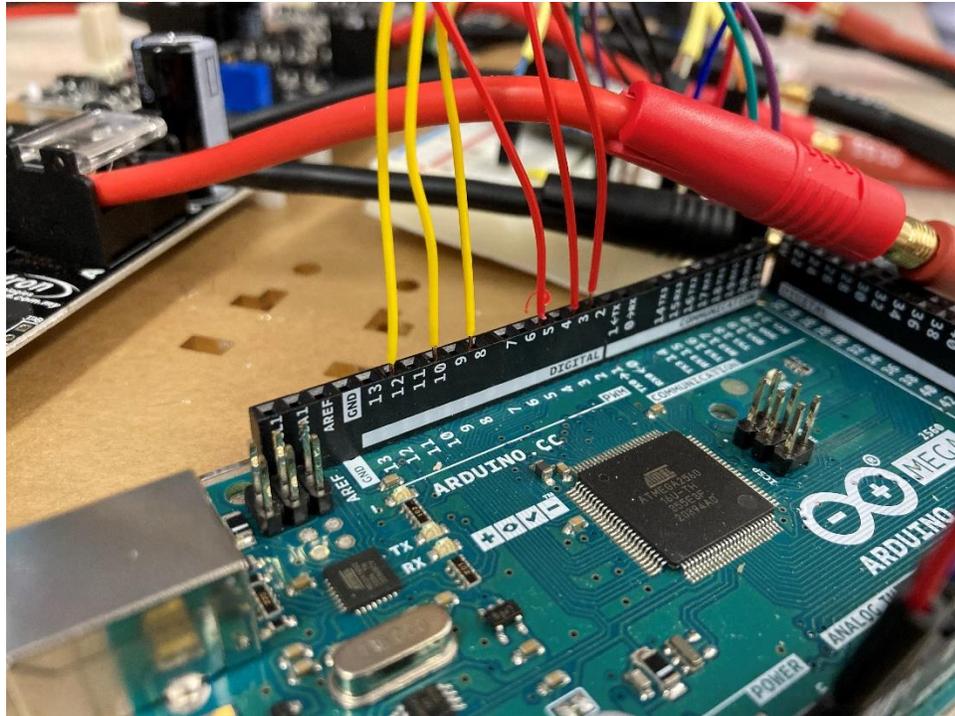
Motor Controllers

To send signals from the Arduino to the motor controllers, each motor controller must be connected with three wires. Each motor controller has a three-pin female connector which plus into the motor controller in only one direction. Each connector has three wires, red, yellow, and black. The three black wires are ground and should be connected to the negative rail on the breadboard. The PWM wires are red and the direction wires are yellow and are directly connected to the Arduino. The pins to be used are shown in the following table.

Motor Controller Connections

	Red	Yellow	Black
X Axis	2	8	Negative rail
Y Axis	3	10	Negative rail
Z Axis	5	12	Negative rail

The PWM and direction wires are shown plugged into the Arduino in the following figure.



Arduino Code

The Arduino code is not particularly complex. To get it running, plug in the Arduino and download the Arduino IDE available at <https://www.arduino.cc/en/software>. Open the sketch (provided in text below and on the drive), making sure it's saved in a folder with the same name. Set the correct Arduino type and serial port (usually COM3 or COM4) in the tools tab. In addition, the sensor libraries must be installed. Under tools, use the library manager to search for “Adafruit INA260”, “Adafruit LIS3MDL”, and “Adafruit LSM6DS”, and install these libraries. They will install necessary dependencies. Next, copy in any desired magnetic field time vectors to the arrays at the beginning of the program. The easy to do this is output them as a CSV, transpose to the correct orientation in Excel, open with a text editor such as notepad or notepad++, and directly paste the values, separated with commas, in to the arrays. Make sure there is a closing bracket }

and semicolon at the end. This is obviously a dubious way to accomplish this, but it does work. Note that the Arduino has limited storage space, necessitating the storage of these arrays in program memory rather than variable memory, which adds some additional syntax. Next, upload the code to the Arduino using the upload button.

It is advisable to check the Arduino serial monitor (under tools, use correct baud rate of 115200) to ensure that each sensor is being detected and the output is not being corrupted or decoded incorrectly. Once this is established, the Arduino IDE can be closed, and the Python program can be used.

Ensure that the Arduino is connected directly to a USB-A port. Connecting through a USB-C adapter was found to not work.

At the time of this writing, outputs from the IMU and current sensors were ignored due to stability issues. Additionally, the current sensors were bypassed with the wiring from the coils directly connecting to the motor controllers. The current sensors were still connected to power and I2C however. Future work should look into improving the Arduino code so that these sensors can be used without issue.

Additional improvements to the Arduino code controlling the PWM and direction outputs should be made. Refer to the MQP report Helmholtz Cage testing section 10.8 for more information.

Python

A Python program was created to provide file-writing functionality as well as better graphing. This was a more difficult problem than initially expected due to the high speed of the serial port and the low speed of the graph, but this program was created to handle these tasks in

multiple threads. Several external libraries are required: PySerial, numpy, pyqtgraph, and pyqt5. These libraries can be installed from the command line with `pip install [name]`.

The purpose of the Python program is twofold: first, to write all the measured data to a file. A uniquely named file is created each time the program is run and will record all measured data as well as the time since start. The user is prompted for the correct serial port.

Second, to allow for real-time graphing. This is useful to visually confirm that the Helmholtz cage is functioning as expected.

Appendix 1: Arduino Code

```
//Do imports
#include <Wire.h>
#include <Adafruit_Sensor.h>
#include <Adafruit_LSM6DSOX.h>
#include <Adafruit_LIS3MDL.h>
#include <Adafruit_INA260.h>
#include <stdbool.h>
//Define each sensor
//No idea why the current ones are different, it's from the example program
Adafruit_LSM6DSOX sox;
Adafruit_LIS3MDL lis3mdl_IMU;
Adafruit_LIS3MDL lis3mdl;
Adafruit_INA260 ina260 = Adafruit_INA260();
Adafruit_INA260 ina260_2 = Adafruit_INA260();
Adafruit_INA260 ina260_3 = Adafruit_INA260();

#define pX 2
#define pY 3
#define pZ 5
#define dX 8
#define dY 10
#define dZ 12

//The basic way to do a loop is by using the delay function, but that's apparently a bad way to do it. These are how you do this different method.
unsigned long current_time;
unsigned long prev_time;
unsigned long start_time;

//Use this time step. The program works with 10. But actually, it will take more than 10 ms to run, and it's not constant.
const unsigned long time_step = 10;

//Do you want to use the extra magnetometer, or just the one integrated into the IMU?
bool useSoloMag = false;

const PROGMEM float x[] = {};
const PROGMEM float y[] = {};
const PROGMEM float z[] = {};

int len = sizeof x / sizeof x[0];

float displayFltX;
float displayFltY;
float displayFltZ;

float PWM_x;
float PWM_y;
float PWM_z;

long k;

//PWM Output scaling
float C_x;
float C_y;
float C_z;

//Direction ints
int dir_x;
int dir_y;
int dir_z;

//Previous direction
int dir_xp;
int dir_yp;
int dir_zp;
```

```

//Direction flip tracking
int n_x = 1;
int n_y = 1;
int n_z = 1;

//Initial Magnetic field
float Bi_x;
float Bi_y;
float Bi_z;

//Generated magnetic field
float B_gen_x;
float B_gen_y;
float B_gen_z;

//The setup loop, to run once when connection is initialized.
void setup(void) {
  //If you need to change the sensor ranges, modes, or data rates, see the examples for the sensors (File->Examples->Examples From Custom
  Libraries) and copy a line in AFTER the sensor is started.

  TCCR3B = TCCR3B & B11111000 | B00000010; // for PWM frequency of 3921.16 Hz

  pinMode(pX, OUTPUT);
  pinMode(pY, OUTPUT);
  pinMode(pZ, OUTPUT);
  pinMode(dX, OUTPUT);
  pinMode(dY, OUTPUT);
  pinMode(dZ, OUTPUT);

  analogWrite(pX, 0);
  analogWrite(pY, 0);
  analogWrite(pZ, 0);
  digitalWrite(dX, LOW);
  digitalWrite(dY, LOW);
  digitalWrite(dZ, LOW);

  //Begin serial connection
  Serial.begin(115200);

  //Tries to start the IMU, prints on fail.
  if (!I2C.begin_I2C()) {
    Serial.println("Failed to find LSM6DSOX chip");
    while (1) {
      delay(10);
    }
  }

  //Tries to start the magnetometer on the IMU
  if (!I2C.begin_I2C(0x1E)) {
    Serial.println("Failed to find LIS3MDL chip");
    while (1) {
      delay(10);
    }
  }

  //Some IMU config
  I2C.setInterruptThreshold(500);
  I2C.configInterrupt(false, false, // enable z axis
    true, // polarity
    false, // don't latch
    true); // enabled!

  //If you are using the other magnetometer too set it up
  if (useSoloMag) {
    if (!I2C.begin_I2C()) {
      Serial.println("Failed to find LIS3MDL chip");
      while (1) {

```

```

    delay(10);
  }
}

lis3mdl.setIntThreshold(500);
lis3mdl.configInterrupt(false, false, true, // enable z axis
                        true, // polarity
                        false, // don't latch
                        true); // enabled!
}

//Start each current sensor with the right address
if (!ina260.begin()) {
  Serial.println("Couldn't find INA260 chip");
  while (1) {
    delay(10);
  }
}
if (!ina260_2.begin(0x41)) {
  Serial.println("Couldn't find INA260 chip");
  while (1) {
    delay(10);
  }
}
if (!ina260_3.begin(0x44)) {
  Serial.println("Couldn't find INA260 chip");
  while (1) {
    delay(10);
  }
}

//PWM correction initial values
float C_x = 1;
float C_y = 1;
float C_z = 1;

//Direction ints
int dir_x = 1;
int dir_y = 1;
int dir_z = 1;

//Previous direction
int dir_xp;
int dir_yp;
int dir_zp;

//Get initial magnetic field measurement
sensors_event_t event_IMU;
lis3mdl_IMU.getEvent(&event_IMU);
Bi_x = event_IMU.magnetic.x*1000;
Bi_y = event_IMU.magnetic.y*1000;
Bi_z = event_IMU.magnetic.z*1000;
Serial.print("Bi_x");
Serial.print(Bi_x);
//Serial.print("Bi_y");
//Serial.print(Bi_y);
//Serial.print("Bi_z");
//Serial.print(Bi_z);

//Set the start time
prev_time = millis();
start_time = millis();
}

//The code that loops constantly
void loop() {
  //get current time
  current_time = millis();

```

```

//If it's time to do an update, do it
//Because this takes so long it'll basically always happen
if (current_time - prev_time >= time_step)
{
  //Print the current time
  Serial.print(current_time);
  Serial.print(",");

  k=(long) (current_time-start_time)/1000+0.5;
  displayFltX = pgm_read_float_near(x + k);
  displayFltY = pgm_read_float_near(y + k);
  displayFltZ = pgm_read_float_near(z + k);

  if (len > k) {
    analogWrite(pX, 0);
    analogWrite(pY, 0);
    analogWrite(pZ, 0);
  }

  B_gen_x = displayFltX - Bi_x;
  B_gen_y = displayFltY - Bi_y;
  B_gen_z = displayFltZ - Bi_z;

  //Serial.print("B gen x");
  //Serial.print(B_gen_x);

  PWM_x = (int) abs(B_gen_x / (C_x * 200000) * 255) + 0.5;
  PWM_y = (int) abs(B_gen_y / (C_y * 200000) * 255) + 0.5;
  PWM_z = (int) abs(B_gen_z / (C_z * 200000) * 255) + 0.5;

  if (PWM_x > 255) {
    PWM_x = 255;
  }
  else if (PWM_x < 0) {
    PWM_x = 0;
  }
  if (PWM_y > 255) {
    PWM_y = 255;
  }
  else if (PWM_y < 4) {
    PWM_y = 0;
  }
  if (PWM_z > 255) {
    PWM_z = 255;
  }
  else if (PWM_z < 4) {
    PWM_z = 0;
  }

  if (B_gen_x >= 0 && n_x >= 0) {
    dir_x = 0;
  }
  else if (B_gen_x < 0 && n_x >= 0) {
    dir_x = 1;
  }
  if (B_gen_y >= 0 && n_y >= 0) {
    dir_y = 1;
  }
  else if (B_gen_y < 0 && n_y >= 0 && PWM_y >= 4) {
    dir_y = 0;
  }
  if (B_gen_z >= 0 && n_z >= 0) {
    dir_z = 1;
  }
  else if (B_gen_z < 0 && n_z >= 0 && PWM_z >= 4) {
    dir_z = 0;
  }
}

```

```

//if (dir_x - dir_xp < 0) {
//n_x = -2000;
//}
//if (dir_y - dir_yp < 0) {
//n_y = -2000;
//}
//if (dir_z - dir_zp < 0) {
//n_z = -2000;
//}

n_x = n_x + 1;
n_y = n_y + 1;
n_z = n_z + 1;

if (n_x > 2) {
  n_x = 1;
}
if (n_y > 2) {
  n_y = 1;
}
if (n_z > 2) {
  n_z = 1;
}

dir_xp = dir_x;
dir_yp = dir_y;
dir_zp = dir_z;

//Serial.print("PWM_x");
//Serial.print(PWM_x);

//Serial.print("dir x");
//Serial.print(dir_x);

//Serial.print("n_x");
//Serial.print(n_x);

analogWrite(pX, PWM_x);
analogWrite(pY, PWM_y);
analogWrite(pZ, PWM_z);

//x A is poritive
if (dir_x == 1) {
  digitalWrite(dX, LOW);
}
if (dir_x == 0) {
  digitalWrite(dX, HIGH);
}
//y A is negative
if (dir_y == 1) {
  digitalWrite(dY, LOW);
}
if (dir_y == 0) {
  digitalWrite(dY, HIGH);
}
//z A is positive
if (dir_z == 1) {
  digitalWrite(dZ, HIGH);
}
if (dir_z == 0) {
  digitalWrite(dZ, LOW);
}

//Get the sensor events, the current sensors do something different down below
sensors_event_t accel;
sensors_event_t gyro;
sensors_event_t temp;

```

```

sox.getEvent(&accel, &gyro, &temp);

//IMU magnetometer
sensors_event_t event_IMU;
lis3mdl_IMU.getEvent(&event_IMU);

//Other magnetometer
sensors_event_t event;
lis3mdl.getEvent(&event);

if (PWM_x < 4) {
    Bi_x = event_IMU.magnetic.x*1000;
}
if (PWM_y < 4) {
    Bi_y = event_IMU.magnetic.y*1000;
}
if (PWM_z < 4) {
    Bi_z = event_IMU.magnetic.z*1000;
}

if (displayFltX <= 1500 && displayFltX >= -1500) {
    C_x = (float) C_x + ((Bi_x - event_IMU.magnetic.x*1000) / Bi_x - 1)/100;
}
else if (displayFltX <= 4500 && displayFltX >= -4500 && event_IMU.magnetic.x*1000 / displayFltX < 0) {
    C_x = C_x;
}
else if (displayFltX / Bi_x > 0 && abs(displayFltX) <= abs(Bi_x)) {
    C_x = (float) C_x - (event_IMU.magnetic.x*1000 / displayFltX - 1)/100;
}
else if (abs(displayFltX - Bi_x) < 4500 && (event_IMU.magnetic.x*1000 - Bi_x) / Bi_x < 0) {
    C_x = C_x;
}
else {
    C_x = (float) C_x + (event_IMU.magnetic.x*1000 / displayFltX - 1)/100;
}
if (displayFltY <= 1500 && displayFltY >= -1500) {
    C_y = (float) C_y + ((Bi_y - event_IMU.magnetic.y*1000) / Bi_y - 1)/100;
}
else if (displayFltY <= 4500 && displayFltY >= -4500 && event_IMU.magnetic.y*1000 / displayFltY < 0) {
    C_y = C_y;
}
else if (displayFltY / Bi_y > 0 && abs(displayFltY) <= abs(Bi_y)) {
    C_y = (float) C_y - (event_IMU.magnetic.y*1000 / displayFltY - 1)/100;
}
else if (abs(displayFltY - Bi_y) < 4500 && (event_IMU.magnetic.y*1000 - Bi_y) / Bi_y < 0) {
    C_y = C_y;
}
else {
    C_y = (float) C_y + (event_IMU.magnetic.y*1000 / displayFltY - 1)/100;
}
if (displayFltZ <= 1500 && displayFltZ >= -1500) {
    C_z = (float) C_z + ((Bi_z - event_IMU.magnetic.z*1000) / Bi_z - 1)/100;
}
else if (displayFltZ <= 4500 && displayFltZ >= -4500 && event_IMU.magnetic.z*1000 / displayFltZ < 0) {
    C_z = C_z;
}
else if (displayFltZ / Bi_z > 0 && abs(displayFltZ) <= abs(Bi_z)) {
    C_z = (float) C_z - (event_IMU.magnetic.z*1000 / displayFltZ - 1)/100;
}
else if (abs(displayFltZ - Bi_z) < 4500 && (event_IMU.magnetic.z*1000 - Bi_z) / Bi_z < 0) {
    C_z = C_z;
}
else {
    C_z = (float) C_z + (event_IMU.magnetic.z*1000 / displayFltZ - 1)/100;
}

if (C_x <= 0.02) {

```

```

    C_x = 1;
}
if (C_y <= 0.02) {
    C_y = 1;
}
if (C_z <= 0.02) {
    C_z = 1;
}
if (C_x >= 1.5) {
    C_x = 1;
}
if (C_y >= 1.5) {
    C_y = 1;
}
if (C_z >= 1.5) {
    C_z = 1;
}
if (PWM_x < 4) {
    C_x = 1;
}
if (PWM_y < 4) {
    C_y = 1;
}
if (PWM_z < 4) {
    C_z = 1;
}

//Serial.print("C x");
//Serial.print(C_x);
//Serial.print("C y");
//Serial.print(C_y);
//Serial.print("C z");
//Serial.print(C_z);

//Print out all the states
//Order is important in the Python code
//Serial.print(accel.acceleration.x);
Serial.print('0');
Serial.print(",");
//Serial.print(accel.acceleration.y);
Serial.print('0');
Serial.print(",");
//Serial.print(accel.acceleration.z);
Serial.print('0');
Serial.print(",");
//Serial.print(gyro.gyro.x);
Serial.print('0');
Serial.print(",");
//Serial.print(gyro.gyro.y);
Serial.print('0');
Serial.print(",");
//Serial.print(gyro.gyro.z);
Serial.print('0');
Serial.print(",");
Serial.print(event_IMU.magnetic.x);
Serial.print(",");
Serial.print(event_IMU.magnetic.y);
Serial.print(",");
Serial.print(event_IMU.magnetic.z);
Serial.print(",");

//If you have the other mag print it. If not, print zeros.
//This makes the Python code not have to handle each case.
if (useSoloMag) {

    Serial.print(event.magnetic.x);
    Serial.print(",");
    Serial.print(event.magnetic.y);

```

```

    Serial.print(",");
    Serial.print(event.magnetic.z);
    Serial.print(",");
}
else {
    Serial.print(0);
    Serial.print(",");
    Serial.print(0);
    Serial.print(",");
    Serial.print(0);
    Serial.print(",");
}

//Print the current sensor info
//Serial.print(ina260.readBusVoltage());
Serial.print('0');
Serial.print(",");
//Serial.print(ina260.readCurrent());
Serial.print('0');
Serial.print(",");
//Serial.print(ina260.readPower());
Serial.print('0');
Serial.print(",");
//Serial.print(ina260_2.readBusVoltage());
Serial.print('0');
Serial.print(",");
//Serial.print(ina260_2.readCurrent());
Serial.print('0');
Serial.print(",");
//Serial.print(ina260_2.readPower());
Serial.print('0');
Serial.print(",");
//Serial.print(ina260_3.readBusVoltage());
Serial.print('0');
Serial.print(",");
//Serial.print(ina260_3.readCurrent());
Serial.print('0');
Serial.print(",");
//Serial.print(ina260_3.readPower());
Serial.print('0');

Serial.print(",");
Serial.print(displayFltX);
Serial.print(",");
Serial.print(displayFltY);
Serial.print(F(","));
Serial.println(displayFltZ);
//Increment time
prev_time = current_time;
}
}

```

Appendix 2: Python Code

```

#William Cooley
#wrcooley@wpi.edu
#Feel free to contact me if you have any issues!

#This code takes in data from the serial port and prints it to a file and plots it live.
#Multithreading is required because the plotting is slow and the serial port is fast
#See the bottom for input arguments including file to output to

#Necessary to install PySerial, numPY, pyqtgraph, pyqt5
#from command prompt use "pip install [name]"
#This may be an issue on WPI managed computers so you may be able to install manually?

```

```

import sys
import threading
import time
import datetime

import serial
import numpy as np
import pyqtgraph as pg
from pyqtgraph.Qt import QtGui, QtCore

#Adapted from:
#https://nrecursions.blogspot.com/2019/09/realtime-plotting-in-python.html
#This class represents the graph that will be displayed. It has two methods:
# __init__ is run once, and update()#is run frequently.
class Graph():
    def __init__(self, datW):
        #Graph set up stuff from the source
        self.app = QtGui.QApplication([])

        #Define a graph window
        #More than one window made the program intolerably slow
        self.IMUandMags = pg.GraphicsWindow()

        #Put the graphs in the window, define their position
        #This is very simple to reconfigure however you like
        self.Acc = self.IMUandMags.addPlot(row=0,col=0)
        self.Gyro = self.IMUandMags.addPlot(row=1,col=0)
        self.Mags = self.IMUandMags.addPlot(row=2,col=0)
        self.TrueMags=self.IMUandMags.addPlot(row=3,col=0)
        self.Currents=self.IMUandMags.addPlot(row=0,col=1)
        self.Voltages=self.IMUandMags.addPlot(row=1,col=1)
        self.Powers=self.IMUandMags.addPlot(row=2,col=1)

        #No idea what this does but it was in the example code
        self.IMUandMags.nextRow()

        #Set up the number of curves you want and which plot to put them on.
        #3 accelerometer
        self.curve1 = self.Acc.plot()
        self.curve2 = self.Acc.plot()
        self.curve3 = self.Acc.plot()

        #3 gyro
        self.curve4 = self.Gyro.plot()
        self.curve5 = self.Gyro.plot()
        self.curve6 = self.Gyro.plot()

        #3 IMU magnetometer
        self.curve7 = self.Mags.plot()
        self.curve8 = self.Mags.plot()
        self.curve9 = self.Mags.plot()

        #3 separate magnetometer
        self.curve10 = self.TrueMags.plot()
        self.curve11 = self.TrueMags.plot()
        self.curve12 = self.TrueMags.plot()

        #9 total current sensor plots (3*3)
        self.curve13 = self.Voltages.plot()

```

```

self.curve14 = self.Currents.plot()
self.curve15 = self.Powers.plot()
self.curve16 = self.Voltages.plot()
self.curve17 = self.Currents.plot()
self.curve18 = self.Powers.plot()
self.curve19 = self.Voltages.plot()
self.curve20 = self.Currents.plot()
self.curve21 = self.Powers.plot()

#Desired magnetic field data
self.curve22 = self.TrueMags.plot()
self.curve23 = self.TrueMags.plot()
self.curve24 = self.TrueMags.plot()

#Graph update speed
#10 ms works well. 100 ms does not.
graphUpdateSpeedMs = 10

#More things from source
#to create a thread that calls a function at intervals
timer = QtCore.QTimer()

#the update function keeps getting called at intervals
#You need the lambda in there! Otherwise it thinks the type is None for some reason.
timer.timeout.connect(lambda:self.update(datW))
timer.start(graphUpdateSpeedMs)
QtGui.QApplication.instance().exec_()

#The method that gets called to update the graph
def update(self,datW):
    #The data is stored in a numpy array which is stored in a tuple. This gets it out
    dat=datW[0]
    #Set the curve data to the desired data. First argument should always be dat[0,:] which is the time vector.
    #Other than that index the self.dat matrix with the same order of variables as in the arduino code
    #Accelerometer
    time=dat[0,:]
    self.curve1.setData(time,dat[1,:])
    self.curve2.setData(time,dat[2,:])
    self.curve3.setData(time,dat[3,:])

    #Gyro
    self.curve4.setData(time,dat[4,:])
    self.curve5.setData(time,dat[5,:])
    self.curve6.setData(time,dat[6,:])

    #IMUMag
    self.curve7.setData(time,dat[7,:])
    self.curve8.setData(time,dat[8,:])
    self.curve9.setData(time,dat[9,:])

    #Standalone Mag
    self.curve10.setData(time,dat[10,:])
    self.curve11.setData(time,dat[11,:])
    self.curve12.setData(time,dat[12,:])

    #Current sensors
    self.curve13.setData(time,dat[13,:])
    self.curve14.setData(time,dat[14,:])
    self.curve15.setData(time,dat[15,:])
    self.curve16.setData(time,dat[16,:])

```

```

self.curve17.setData(time,dat[17,:])
self.curve18.setData(time,dat[18,:])
self.curve19.setData(time,dat[19,:])
self.curve20.setData(time,dat[20,:])
self.curve21.setData(time,dat[21,:])

#Intended magnetic field
self.curve22.setData(time,dat[22,:])
self.curve23.setData(time,dat[23,:])
self.curve24.setData(time,dat[24,:])

#Make stuff happen
self.app.processEvents()

#This class will be run by the plotting thread
def plotting(datW,run_event):
    #Discussed lower, but this should stop the program when this flag is false
    while run_event.is_set():
        g = Graph(datW)

#This class will be run by the collecting thread
def collecting(datW,linesToSkip,connection,out,run_event):
    #Used to skip first few lines because they often contain garbage data and break the decode line.
    lineNumber=0
    while lineNumber<linesToSkip:
        #Get the raw message out of the serial buffer and increment the counter
        rawMsg=connection.readline()
        lineNumber+=1;

#Discussed lower, but this should stop the program when this flag is false
while run_event.is_set():
    #Get current line
    rawMsg=connection.readline()

    #Parse message into list of numbers and print
    #Sometimes a data packet is lost or corrupted, so this is in a try: except: block just in case
    #This complicated one-liner takes the raw message in bytes, then decodes it to UTF-8, then removes whitespace and newline
    #characters, then splits up into a list based on commas. Then it takes this list of strings and turns it into a numpy array
    #of floating-point numbers, then it transposes it from a row to a column.
    #This could probably be significantly improved by directly translating the bytestring into the numpy array
    try:
        nums=np.transpose(np.array(list(map(float,list(rawMsg.decode('utf8')).strip().split(','))))))
        #This is how the data shifts over time. Copy the array into itself but one spot to the left (ditching the first column).
        dat[:,:-1]=dat[:,1:]

        #Then change the last column to the newly found data
        dat[:,:-1]=nums
        datW=(dat,)

        for i in nums:
            out.write(str(i))
            out.write(" ")
        out.write("\n")

    except Exception as e:
        print(e)
        print("Numerical error")

#This is what runs
if __name__ == '__main__':

```

```

#Define the number of states you are using.
#Time is 1
#IMU+Mag integrated chip adds 9
#Standalone Magnetometer adds 3
#Each current sensor adds 3
#With all sensors, and desired field outputs, numStates=22
#With output of desired magnetic field at the end as well, 25
numStates=25

#These are necessary to omit the first 5 lines. They can contain leftover data from the previous serial transmission
#And break the encoding
linesToSkip=5

#max number of data points to show on graph
#Proportional to time
maxLen = 1000

#Create empty numpy array of correct dimensions
#Numpy arrays are fast which is necessary for this
dat=np.zeros(shape=(numStates,maxLen))

#You have to box up the array inside this tuple of 1, because otherwise python interprets each array column as an argument to
#the thread below. It's odd, but this is the best way I found around that.
datW=(dat,)

#Why is multithreading necessary? Data needs to be collected from the serial port very quickly. If the program gets behind,
#issues can occur and the graph looks jumpy. But, the graphing is pretty slow. Too slow to graph every time new data is found.

#To get around this, we have two separate threads running, one which collects the data and writes to an array, and which reads
#that array and plots it. The collection can be very quick, and the graphing can be slow, and this works because they are allowed
#to operate at different rates with the threading.

#However it's hard to kill threads. So this flag is defined that can be seen across threads and will end them.
run_event=threading.Event()
run_event.set()

#Open the files in this context rather than inside the thread
#Start the serial connection and open the file. Doing it like this should automatically close them when the program is interrupted
#or the graph window is closed.
#If you want to append instead of overwrite the file, change the 'wb+' into 'ab+'.
print("Press control-c to exit the program once it has started.")
port=input("Please enter full serial port name, example: COM3: ")
try:
    with serial.Serial(port,115200) as connection, open("Auto-
"+datetime.datetime.now().strftime("%Y_%m_%d_%H_%M_%S"+'.txt'),'w+') as out:
        #Create the threads, pass in the necessary arguments
        DataCollectionThread=threading.Thread(target=collecting,args=(datW,linesToSkip,connection,out,run_event))
        PlottingThread=threading.Thread(target=plotting,args=(datW,run_event))

        #Start the threads
        DataCollectionThread.start()
        PlottingThread.start()

        #Block necessary to be able to keyboardinterrupt and stop script
        try:
            while True:
                time.sleep(1)
        except (KeyboardInterrupt,SystemExit):
            print("Exiting program")
            do_run.clear()

```

```
DataCollectionThread.join()
PlottingThread.join()
connection.close()
out.close()
time.sleep(10)
sys.exit()
print("File and port successfully closed")
except serial.serialutil.SerialException:
    print("Serial Port not found,closing in 10 sec")
    time.sleep(10)
    sys.exit()
except Exception as e:
    print(e)
    time.sleep(10)
```